EPSON®

EXCEED YOUR VISION

S1D13513 Display Controller

Hardware Functional Specification

Document Number: X78C-A-001-01

Status: Revision 1.7

Issue Date: 2010/07/28

Information in this document is subject to change without notice. You may download and use this document, but only for your own use in evaluating Seiko Epson/EPSON products. You may not modify the document. SEIKO Epson, Inc. disclaims any representation that the contents of this document are accurate or current. The Programs/Technologies described in this document may contain material protected under U.S. and/or International Patent laws.

Table Of Contents

1 Introduction

1.1 Scope

This is the Hardware Functional Specification for the S1D13513 LCD Display Controller. Included in this document are timing diagrams, AC and DC characteristics, register descriptions, and power management descriptions. This document is intended for two audiences: Video Subsystem Designers and Software Developers.

This document is updated as appropriate. Please check for the latest revision of this document before beginning any development. The latest revision can be downloaded at www.erd.epson.com.

We appreciate your comments on our documentation. Please contact us via email at documentation@erd.epson.com.

1.2 Overview Description

The S1D13513 is a highly integrated Display Controller capable of outputting to LCD or TV. With the flexibility of an external SDRAM memory interface, this low cost, low power, device supports a wide range of CPUs, panels, and a camera port that can be configured as 2x 8-bit ports. The S1D13513 feature set and architecture are designed to meet the requirements of embedded systems such as Mobile Communications, Hand-Held PC's, Office Automation, and Automotive applications.

The S1D13513 features both Sprite and 2D BitBLT engines designed to reduce the load on the Host, while increasing the performance of graphics intensive operations. Additionally, the S1D13513 offers such features as multiple windows, alpha blending, gamma correction, and mirror/rotation function which allow user configurability of various images on the Main/PIP1/PIP2 displays. While focusing on devices targeted by the Microsoft Windows CE Operating System, the S1D13513's impartiality to CPU type or operating system makes it an ideal display solution for a wide variety of applications.

1.3 Package Limitations

The S1D13513 is available in two package formats: PBGA 256-pin and QFP22 208-pin. However, the QFP package has less pins and does not support the full feature set of the S1D13513. The following features are not available on the QFP package of the S1D13513.

- x32 SDRAM is not supported (x16 SDRAM only)
- 256Mbit (32M byte) and 512Mbit (64M byte) SDRAM sizes are not supported
- 24-bit TFT is not supported (18-bit TFT maximum)
- Camera2 interface is not available (only Camera1 interface multiplexed with YUV Digital Out)
- Keypad Interface is not available
- INT2# is not supported (INT1# only)

2 Features

2.1 Memory

- Uses external SDRAM or mobile SDRAM as the display buffer (no embedded memory)
  - External memory can be used for certain 2D BitBLT and Sprite functions
  - Memory is addressable using direct or indirect access modes
  - Linear access to the first 1M bytes of memory and four configurable 256K byte windows into the remaining area
- SDRAM Interface
  - Supports up to 100MHz SDRAM bus clock
  - Supports x16 and x32 SDRAM interfaces (x32 only available for PBGA package)
  - Supports 8/16/32/64M bytes of 4 bank SDRAM (32/64M bytes only supported for PBGA package)
  - Supports 8/16/32/64M bytes of 4 bank Mobile SDRAM (32/64M bytes only supported for PBGA package)
  - Low power design
  - Automatic re-entry into self refresh mode

Note

For memory usage guidelines, see Section 18.2, “Memory Bandwidth” on page 428.

2.2 CPU Interfaces

- Direct and indirect interface support of the following 16-bit CPU interfaces:
  - Generic MPU bus interface
  - Renesas SH-4 / SH-3
  - MIPS/ISA bus interface
  - FreeScale MC68K Bus1 type interface
  - FreeScale MC68K Bus2 type interface
  - FreeScale PowerPC bus interface
  - Philips PR31500/PR31700 (16-bit memory accesses only)
  - Toshiba TX3912 (16-bit memory accesses only)
  - FreeScale MPC555 with Burst Transfer

- Serial Host Interface
- Registers are memory-mapped - M/R# input selects between memory and register address space

2.3 Panel Interface Support

- RGB interface single-panels
  - Color TFT Panels
    - 16/18/24-bit interface (24-bit panels are not supported for QFP package)
    - Generic TFT/TFD interface
    - HR-TFT interface
  - Passive Panels
    - 8-bit Monochrome (PIP2 window is not supported for passive panels)
    - 8-bit Color Type 2 (PIP2 window is not supported for passive panels)
    - Passive panels have additional maximum display size limitations. Please contact your EPSON representative for details.
- Optional serial command interface supports:
  - TFT w/µ-Wire interface (16-bit)
  - EPSON ND-TFD 4 pin interface (8-bit)
  - EPSON ND-TFD 3 pin interface (9-bit)
  - 8/24-bit command interface
- Panel Resolution Examples
  - For memory usage guidelines, see Section 18.2, “Memory Bandwidth” on page 428.
- YUV Digital Output (YUV 4:2:2) which supports NTSC/PAL format TV via an external video encoder (i.e. ADV7170)

2.4 Display Features

- Multiple window (layer) support
    - Main window and PIP1 window (if enabled) form the View Port (bottom layer)
        - 8/16/32 bit-per-pixel (bpp) color depths
        - alpha blending is not supported for the View Port
        - mirror and 180° rotation functions
        - Main or PIP1 window can be double-buffered
        - optional gamma correction
    - PIP2 window (if enabled) is the top layer (PIP2 is not supported for passive panels)
        - 8/16/32 bit-per-pixel (bpp) color depths
        - optional alpha blending function (ARGB 1:5:5:5 / ARGB 4:4:4:4 / ARGB 8:8:8:8 formats including alpha map)
        - optional transparency function
        - mirror and 180° rotation functions
        - PIP2 window can be double-buffered
        - optional gamma correction
- Mirror and Rotation
    - Mirror function performs a horizontal flip of the display image
        - independent controls for View Port (Main/PIP window) and PIP2 window
    - Rotation function performs a 180° counter-clockwise rotation of the display image
        - independent controls for View Port (Main/PIP window) and PIP2 window
- Double-Buffering Support
    - Double-buffered window is selectable from Main/PIP1/PIP2 window (only one window can be double-buffered at a time)
    - Automatically switches between front and back buffer
    - Image data source can be from the Camera interface or the Sprite Engine
- Alpha Blending
    - Supports alpha blending between the View Port (Main + PIP1) and the PIP2 window
    - 8-bit constant alpha value
    - Dynamic alpha with alpha map
    - Combined with transparency

- Gamma Correction
  - Selectable gamma correction for Main/PIP1/PIP2 windows
  - 2 Gamma Correction Look-up Tables (Bank A and Bank B) using Single Port SRAM
  - Independent color correction for each of RGB components
  - View Port (Main+PIP1 window) can be dynamically switched between LUT banks
  - Async table access
- Pseudo Color Expansion
  - Async table access
  - 2x2 Dither matrix
  - 2x2 FRM
  - Error Diffusion
  - 2-bit color reduction
- Hardware cursor is available using the Sprite Engine
- Camera1 or Camera 2 YUV image can be displayed in the PIP1 or PIP2 window
- Interrupt support
  - Maskable Non-Display (Vsync) interrupt
  - Vsync interrupt assertion can be delayed a configurable number of lines

2.5 Clock Source

- 2 embedded PLLs
  - PLL1 source: BUSCLK, CLKI3, OSCI1/OSCO1, or OSCI2/OSCO2
  - PLL2 source: BUSCLK, CLKI3, OSCI1/OSCO1, or OSCI2/OSCO2
- Two crystal inputs: OSCI1/OSCO1 and OSCI2/OSCO2
- Four digital inputs: BUSCLK, CLKI3, CMCLKIN1, CMCLKIN2
- Clock Outputs: MEMCLK, CMCLKOUT1, CMCLKOUT2

2.6 2D Acceleration

- Supports 8/16/32 bit-per-pixel (bpp) color depths
- Linear or Rectangular data transfers for the Source or Destination
- 64-bit 2D BitBLT engine including:

  Write BitBLT w/Color Expansion
  Write BitBLT w/ROP
  Write BitBLT w/Transparency
  Read BitBLT
  Move BitBLT w/Color Expansion
  Move BitBLT w/ROP
  Move BitBLT w/Transparency
  Solid Fill BitBLT
  Pattern Fill BitBLT w/ROP
  Pattern Fill BitBLT w/Transparency

2.7 Sprite Engine

- The Sprite Engine is a 2D animation accelerator which supports:
  - up to 16 sprites, each with up to 16 frames to be displayed in a sequence
  - Z-order value for each sprite which is used for alpha blending and transparency
  - Image rotation and mirror settings specific for each sprite
  - Optional arbitrary rotation settings for each sprite
  - Sprite Source format: RGB 5:6:5, ARGB 1:5:5:5, ARGB 4:4:4:4, ARGB 8:8:8:8
  - Sprite Output formats: RGB 3:3:2, RGB 5:6:5, RGB 8:8:8
  - Sprite display size up to 1280 x 1280
  - Sprite virtual image size up to 8192 x 8192
  - Alpha Blending
  - Up to 256 pattern sequential animation function

2.8 Command FIFO

- Command FIFO is designed to reduce CPU load when issuing BitBLT and Sprite commands
    - Allows up to a maximum of 64 entries
    - BitBLT and Sprite commands can be mixed within the Command FIFO. However, the BitBLT and Sprite Engines cannot be performing operations simultaneously.
    - Command FIFO automatically sends the next command to the BitBLT/Sprite engine as soon as it is available
    - Commands are sent to the FIFO by writing to register space

2.9 Camera Interface

- Dual Port Camera Interface
    - Configurable as two 8-bit camera interfaces or one 16-bit camera interface
    - YUV 4:2:2 input format (8 or 16-bit)
    - Optionally capture raw JPEG data from a JPEG capable camera
    - Supports ITU-R BT656 (CCIR-656) YUV format
    - Camera strobe function
- Dual resizers allow resizing of YUV data for both the Capture and View path
- YUV image data can be “Captured” and stored into SDRAM as YUV 4:2:2 format
- YUV image data can be “Viewed” (or displayed) on the LCD or TV

Note

Camera2 8-bit input and Camera1 16-bit input are not available for QFP package.

2.10 Miscellaneous

- Supports 20-50MHz Host bus clock
- Supports maximum 50MHz Internal System Clock
- Two IRQ output pins (INT2# pin is only available on PBGA package)
  - IRQ Source (VSYNC, Delayed VSYNC, BitBLT done, Sprite Done, Etc.)
- I2C interface (typically used to program the Camera)
- PWM: 4 channel for backlight control
- Keypad Interface (only available on PBGA package)
  - 5 x 5 matrix support
- Software initiated power save mode
- Clocks are dynamically turned off when modules are not needed
- General Purpose Input/Output pins are available
- Operating system independent
- Power Supplies:
  - IO operates at 3.3 volts ± 0.3V
  - Core operates at 1.8 volts ± 0.15V
- Package Types:
  - PBGA 256-pin package (17 x 17 x 1.7 mm, Ball pitch: 1.0 mm)
  - QFP22 208-pin package (28 x 28 x 1.4 mm, Pin pitch: 0.5 mm)

Note

The QFP package does not support all S1D13513 features. For further information, refer to Section 1.3, “Package Limitations” on page 10.

3 System Diagram

The following diagrams are examples of a typical system implementation. For detailed pin descriptions and pin mapping, refer to Section 5, “Pins” on page 21.

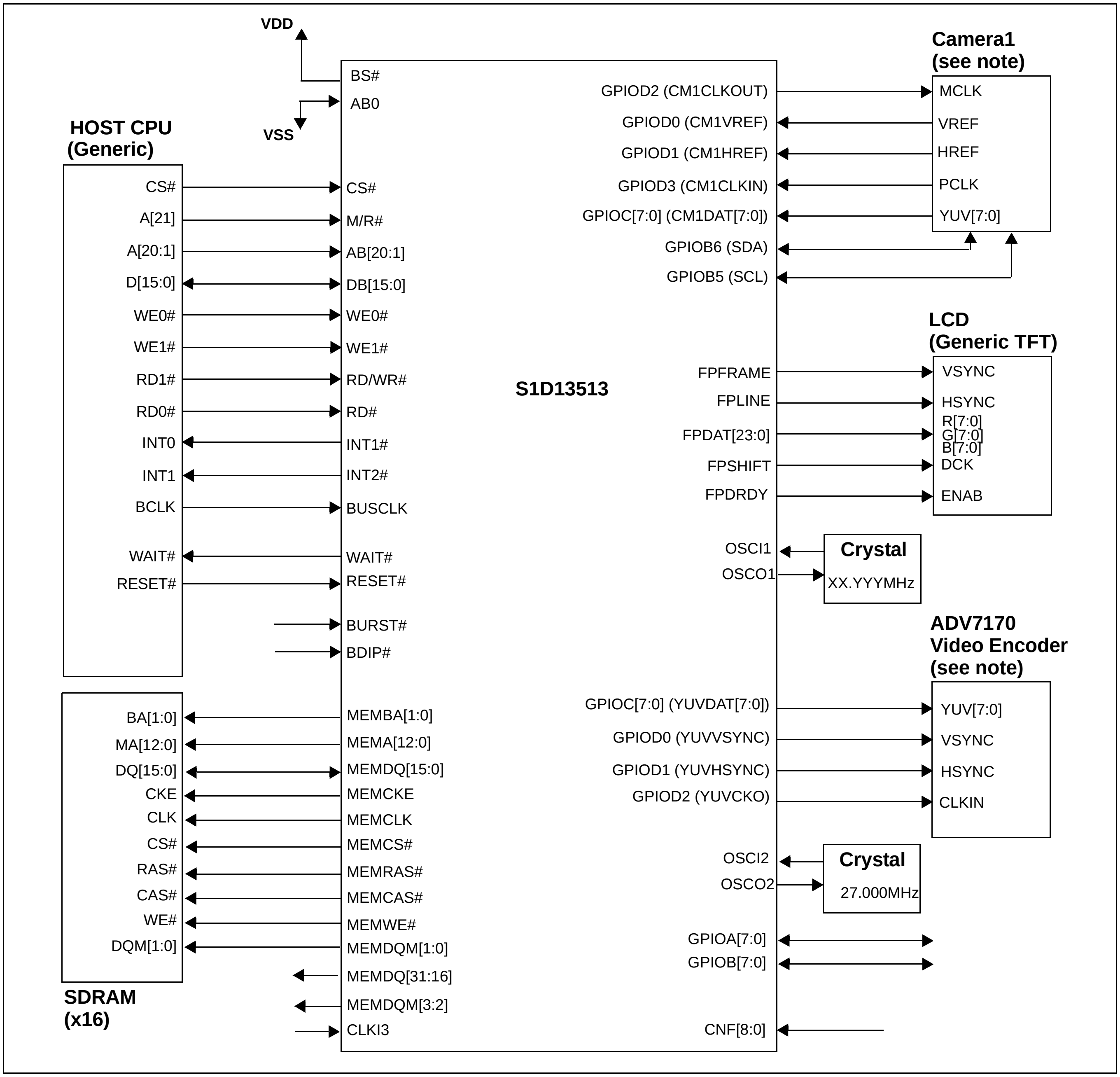


Figure 3-1: Example System Diagram 1

Note

For the above example system diagram, Camera1 and the ADV7170 Video Encoder cannot be active at the same time.

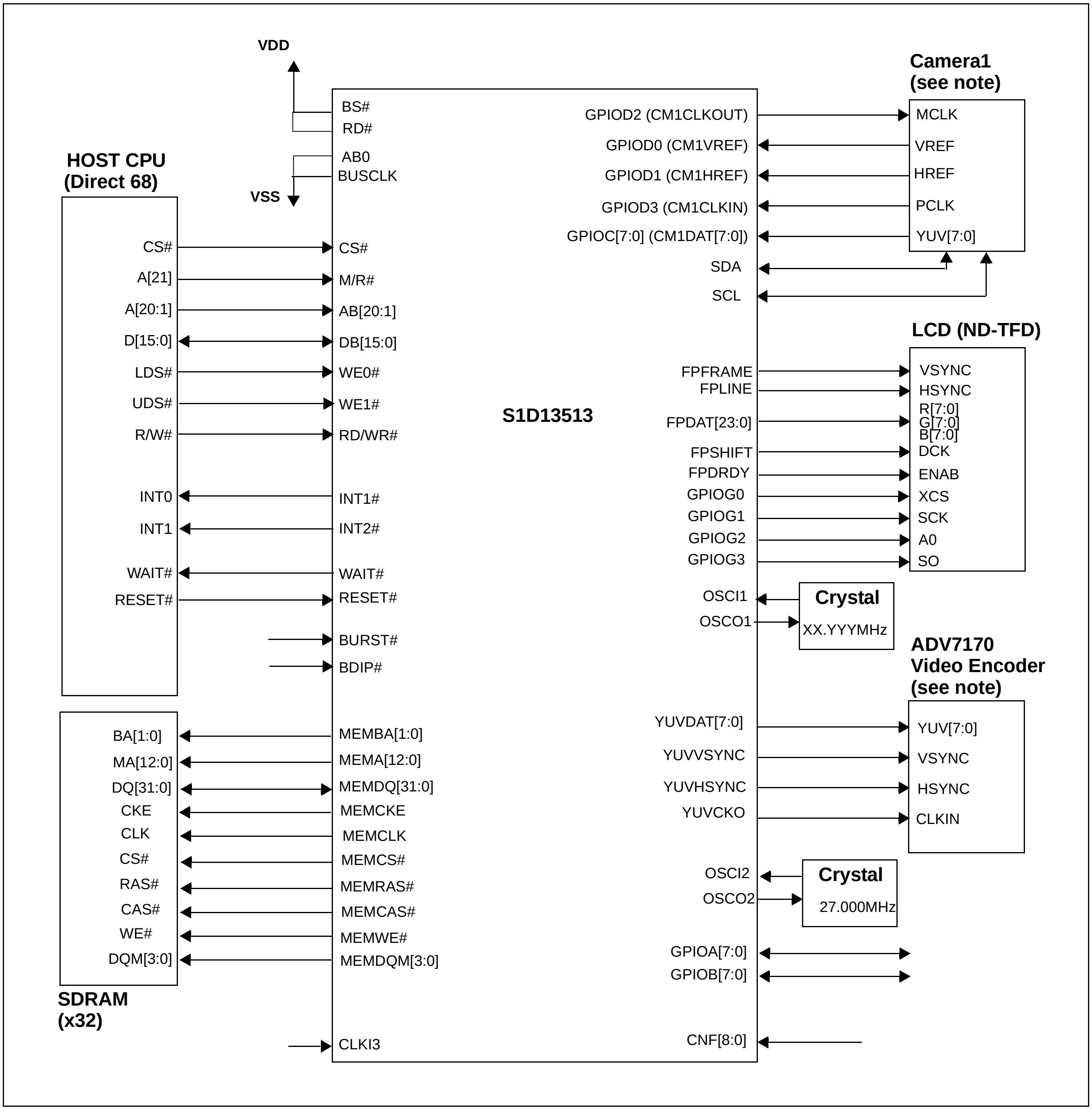


Figure 3-2: Example System Diagram 2

Note

For the above example system diagram, Camera1 and the ADV7170 Video Encoder cannot be active at the same time.

4 Block Diagram

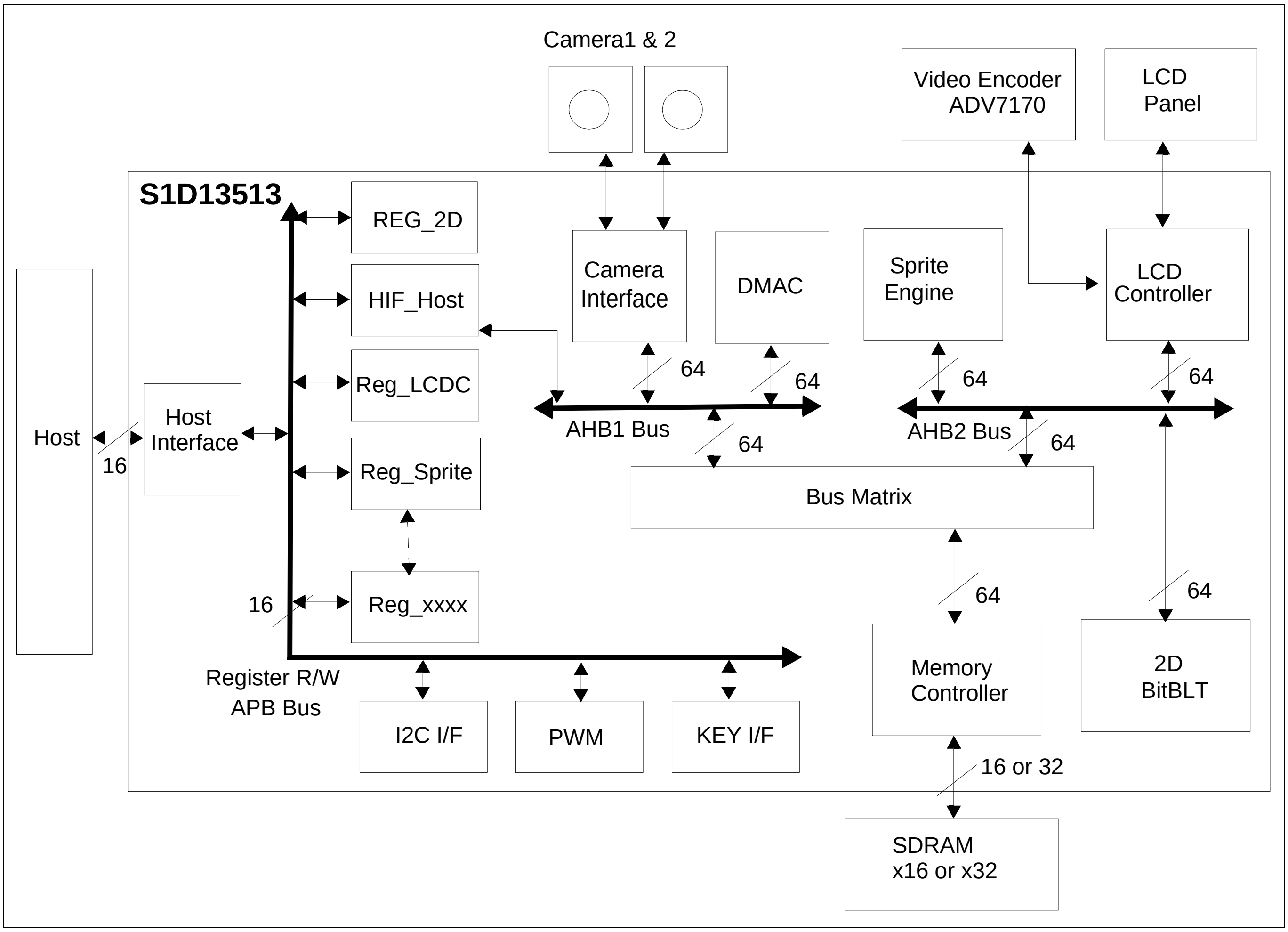


Figure 4-1: Block Diagram

5 Pins

The S1D13513 is available in two package formats:

- PFBGA 256-pin
- QFP22 208-pin

However, the QFP package has less pins and does not support the full feature set of the S1D13513. For a summary of the QFP limitations, see Section 1.3, “Package Limitations” on page 10.

5.1 S1D13513 Pinout Diagrams

5.1.1 QFP22 208-pin Pinout

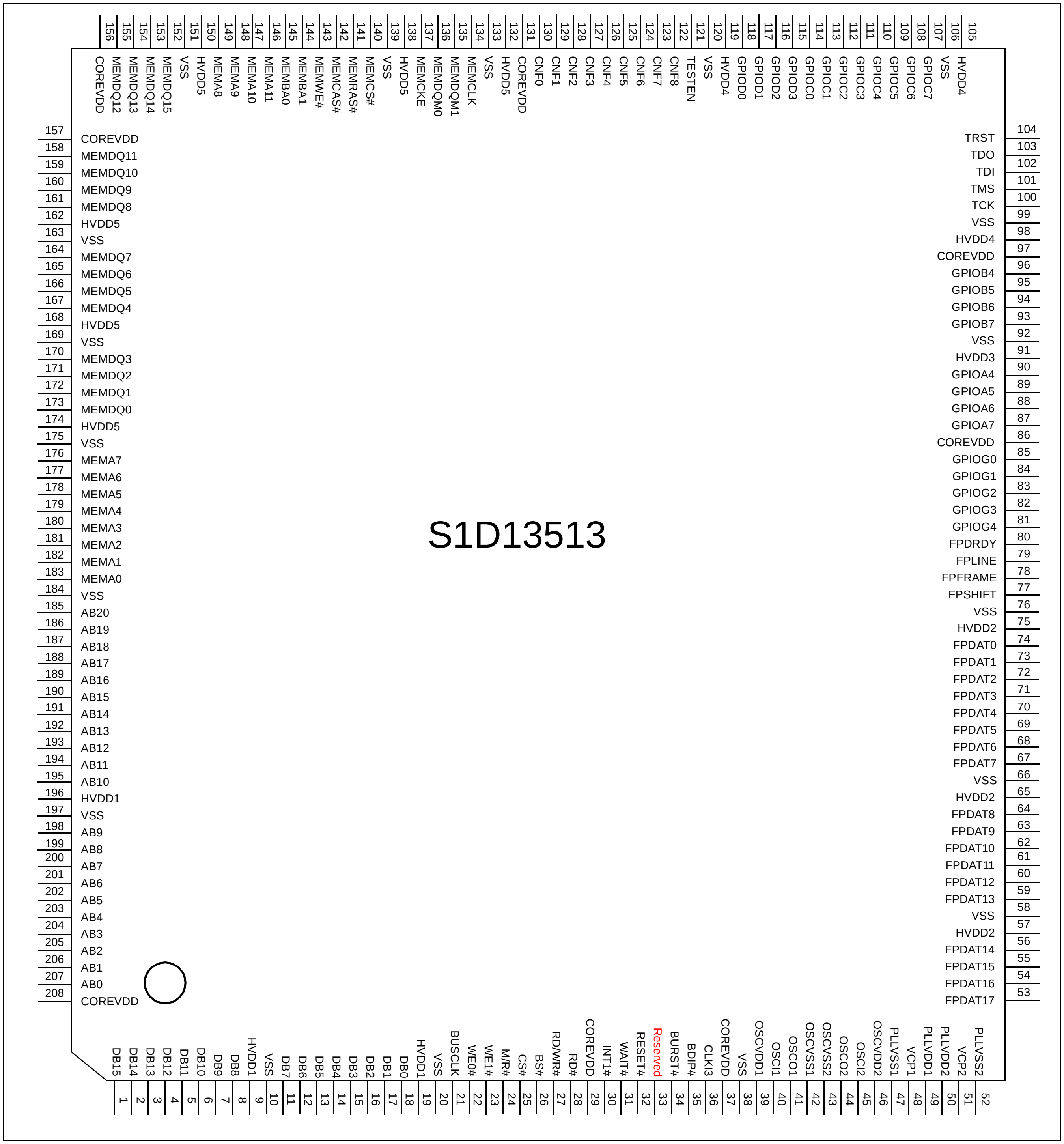


Figure 5-1: QFP22-208 Pinout

5.1.2 PBGA 256-pin Pinout

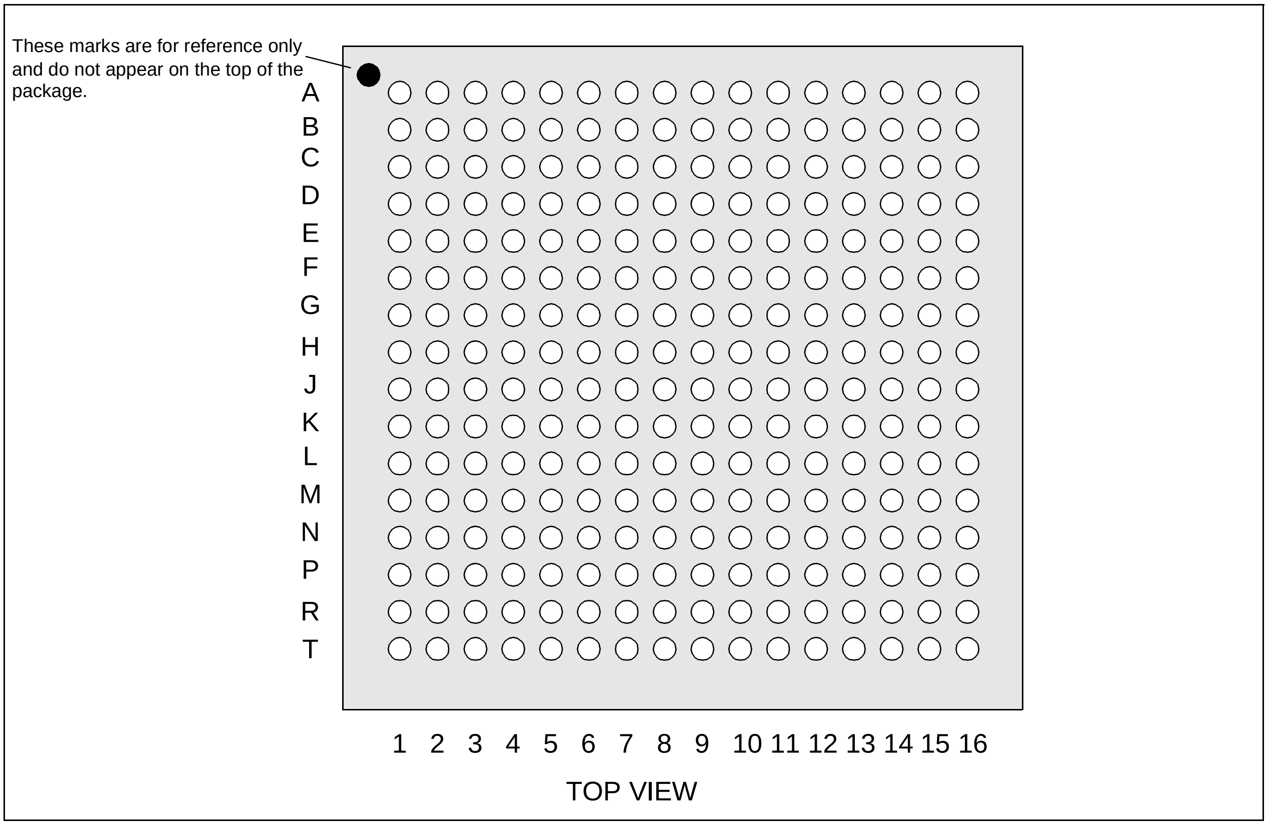

These marks are for reference only and do not appear on the top of the package.
A
B
C
D
E
F
G
H
J
K
L
M
N
P
R
T
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16
TOP VIEW

Figure 5-2: PBGA 256-pin Pinout

Table 5-1 : PBGA 256-pin Pinout

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	
A	VSS	AB1	AB3	AB7	AB8	AB14	AB15	MEMA4	MEMA7	MEMDQ1	MEMDQ22	MEMDQ7	MEMDQ24	MEMDQ10	MEMDQ27	VSS	A
B	AB0	AB2	AB4	AB6	AB9	AB13	AB19	MEMA3	HVDD5	MEMDQ2	MEMDQ4	MEMDQ23	MEMDQ25	MEMDQ26	MEMDQ11	MEMDQ12	B
C	DB15	DB14	COREVDD	AB5	AB10	AB11	AB16	MEMA2	MEMDQ0	MEMDQ3	MEMDQ21	HVDD5	MEMDQ9	COREVDD	MEMDQ28	MEMDQ29	C
D	DB13	DB11	DB12	VSS	HVDD1	AB12	AB18	VSS	MEMDQ16	MEMDQ19	MEMDQ5	MEMDQ8	VSS	MEMDQ13	MEMDQ14	MEMDQ31	D
E	DB7	HVDD1	DB9	DB10	DB8	AB17	AB20	MEMA1	MEMA6	MEMDQ18	MEMDQ20	MEMDQ6	MEMDQ30	MEMDQ15	HVDD5	MEMA8	E
F	DB1	DB3	DB5	DB6	DB4	COREVDD	DB2	MEMA0	MEMA5	MEMDQ17	COREVDD	MEMA9	MEMA10	MEMA11	MEMA12	MEMBA0	F
G	M/R#	BUSCLK	DB0	HVDD1	WE1#	CS#	VSS	VSS	VSS	VSS	HVDD5	MEMBA1	VSS	MEMCAS#	MEMRAS#	MEMWE#	G
H	WE0#	RD/WR#	BS#	RD#	INT1#	INT2#	VSS	VSS	VSS	VSS	MEMCS#	HVDD5	MEMDQM0	MEMDQM1	MEMDQM2	MEMCKE	H
J	VSS	WAIT#	RESET#	BURST#	Reserved	BDIP#	VSS	VSS	VSS	VSS	MEMDQM3	HVDD5	CNF1	CNF3	CNF0	MEMCLK	J
K	CLKI3	VSS	COREVDD	FPDAT23	FPDAT19	FPDAT7	VSS	VSS	VSS	VSS	CNF2	CNF5	VSS	CNF8	CNF6	CNF4	K
L	OSCI1	OSCO1	OSCVSS1	OSCVDD1	HVDD2	COREVDD	FPDAT5	HVDD2	GPIOG0	GPIOA6	COREVDD	HVDD4	GPIOD1	GPIOD2	TESTEN	CNF7	L
M	OSCI2	OSCO2	OSCVDD2	OSCVSS2	FPDAT10	FPDAT6	FPDAT0	FPDRDY	GPIOA7	GPIOA5	GPIOB7	GPIOC1	HVDD4	GPIOC4	GPIOD3	GPIOD0	M
N	VCP1	PLLVSS1	VSS	FPDAT17	FPDAT11	VSS	FPDAT1	GPIOG4	VSS	GPIOA1	GPIOB6	TDI	VSS	GPIOC5	GPIOC3	GPIOC0	N
P	PLLVDD1	PLLVDD2	COREVDD	FPDAT16	FPDAT20	FPDAT9	FPDAT4	FPSHIFT	GPIOG1	GPIOA0	GPIOB1	TCK	TRST	COREVDD	GPIOC6	GPIOC2	P
R	VCP2	PLLVSS2	FPDAT22	FPDAT14	FPDAT18	FPDAT8	FPDAT3	FPFRAME	GPIOG2	GPIOA2	HVDD3	GPIOB2	GPIOB0	TMS	TDO	GPIOC7	R
T	VSS	FPDAT15	FPDAT21	FPDAT13	FPDAT12	HVDD2	FPDAT2	FPLINE	GPIOG3	GPIOA3	GPIOA4	GPIOB3	GPIOB5	GPIOB4	HVDD4	VSS	T
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	

5.2 Pin Description

Key:

Pin Types

I	=	Input
O	=	Output
IO	=	Bi-Directional (Input/Output)
P	=	Power pin

RESET# States

H	=	High level output
L	=	Low level output
Z	=	High Impedance (Hi-Z)
1	=	Pull-up resistor on input
0	=	Pull-down resistor on input
#	=	Active low level

Table 5-2 : Cell Descriptions

Cell	Description
ILTR	Low voltage transparent input
OLTR	Low voltage transparent output
IC	LVCMOS input
ICD1	LVCMOS input with pull-down resistor (50kΩ@3.3V)
ICD2	LVCMOS input with pull-down resistor (100kΩ@3.3V)
ICU1	LVCMOS input with pull-up resistor (50kΩ@3.3V)
ICS	LVCMOS schmitt input
ICSD1	LVCMOS schmitt input with pull-down resistor (50kΩ@3.3V)
ICSP1	LVCMOS schmitt input with pull-up resistor (50kΩ@3.3V)
ICSP2	LVCMOS schmitt input with pull-up resistor (100kΩ@3.3V)
OTLN4	Low noise output buffer (4mA@3.3V, deltaV = 0.4V)
OTLN8	Low noise output buffer (8mA@3.3V, deltaV = 0.4V)
BLNC4D1	Low noise LVCMOS IO buffer (4mA@3.3V) with pull-down resistor (50kΩ@3.3V)
BLNC4D2	Low noise LVCMOS IO buffer (4mA@3.3V) with pull-down resistor (100kΩ@3.3V)
BLNC4P1	Low noise LVCMOS IO buffer (4mA@3.3V) with pull-up resistor (50kΩ@3.3V)
BLNCS4D1	Low noise LVCMOS schmitt IO buffer (4mA@3.3V) with pull-down resistor (50kΩ@3.3V)
P	Power

5.2.1 Host Interface

Many of the host interface pins have different functions depending on the host bus interface that is selected using the CNF[6:0] pins (see Section 5.3, “Summary of Configuration Options” on page 41). To determine the pin mapping and pin functions for each host bus interface, refer to Table 5-11:, “Host Bus Interface Pin Mapping 1 (Direct Interfaces),” on page 44, Table 5-12:, “Host Bus Interface Pin Mapping 2 (Indirect Interfaces),” on page 45, Table 5-13:, “Host Bus Interface Pin Mapping 3 (Direct Interfaces),” on page 46, and Table 5-14:, “Host Bus Interface Pin Mapping 4 (Indirect Interfaces),” on page 47.

Table 5-3 : Host Interface Pin Descriptions

Pin Name	Type	QFP Pin#	PBGA Ball #	Cell	Power	RESET# State	Description
AB[20:0]	I	185-195, 198-207	E7,B7,D7, E6,C7,A7, A6,B6,D6, C6,C5.B5. A5.A4,B4, C4,B3,A3, B2,A2,B1	ICD1	HVDD1	Z	These input pins are System Address pins 20-0. For detailed pin functions for each host bus interface, see Section 5.4, “Host Bus Interface Pin Mapping” on page 44.
DB[15:0]	IO	1-8,11-18	C1,C2,D1, D3,D2,E4, E3,E5,E1, F4,F3,F5, F2,F7,F1,G3	BLNC4D1	HVDD1	Z	These input/output pins are System Data Bus pins 15-0. For detailed pin functions for each host bus interface, see Section 5.4, “Host Bus Interface Pin Mapping” on page 44.
CS#	I	25	G6	ICU1	HVDD1	Z	This input pin is Chip Select and has multiple functions configured according to the Host Bus Interface selected using CNF[4:0]. • For 1 CS# mode, this pin inputs the chip select signal (CS#). • For 2 CS# mode, this pin inputs the memory chip select signal (CSM#). • When the Serial Host Interface is selected, this pin is the Serial Host Chip Select (SCS#). For detailed pin functions for each host bus interface, see Section 5.4, “Host Bus Interface Pin Mapping” on page 44.

Table 5-3 : Host Interface Pin Descriptions

Pin Name	Type	QFP Pin#	PBGA Ball #	Cell	Power	RESET# State	Description
M/R#	I	24	G1	ICU1	HVDD1	Z	This input pin has multiple functions. • For 1 CS# mode, this pin selects between the display buffer and register address spaces. When M/R# is set high, the display buffer is accessed and when M/R# is set low the registers are accessed. • For 2 CS# mode, this pin inputs the register chip select (CSR#). • For Indirect Host Bus Interfaces and Serial Host Bus Interfaces, the internal pull-down resistor is enabled and this pin must be left unconnected. • For Serial Host Bus Interfaces, the internal pull-down resistor is enabled and this pin must be left unconnected. For detailed pin functions for each host bus interface, see Section 5.4, “Host Bus Interface Pin Mapping” on page 44.
RD#	I	28	H4	ICU1	HVDD1	Z	This input pin has multiple functions. • For the Generic interface, this pin is the lower byte read command (RD0#). • For the Direct/Indirect 68 interface, this pin must be connected to VDD. • For the Direct/Indirect 80 Type 1 and Type 2 interfaces, this pin is the read enable signal (RD#). • For detailed pin functions for all other interfaces, refer to the Specification for each Host. For detailed pin functions for each host bus interface, see Section 5.4, “Host Bus Interface Pin Mapping” on page 44.
RD/WR#	I	27	H2	ICU1	HVDD1	Z	This input pin has multiple functions. • For the Generic interface, this pin is the upper byte read command (RD1#). • For the Direct/Indirect 68 interface, this pin is the read/write signal (R/W#). • For the Direct/Indirect 80 Type 1 interface, this pin is the write enable signal (WE#). • For the Direct/Indirect 80 Type 2 interface, this pin must be connected to VDD. • For detailed pin functions for all other interfaces, refer to the Specification for each Host. For detailed pin functions for each host bus interface, see Section 5.4, “Host Bus Interface Pin Mapping” on page 44.

Table 5-3 : Host Interface Pin Descriptions

Pin Name	Type	QFP Pin#	PBGA Ball #	Cell	Power	RESET# State	Description
WE0#	I	22	H1	ICU1	HVDD1	Z	This input pin has multiple functions. • For the Generic interface, this pin is the lower byte write enable signal (WE0#). • For the Direct/Indirect 68 interface, this pin is the lower data strobe (LDS#). • For the Direct/Indirect 80 Type 1 interface, this pin is the lower byte enable signal (LBE#). • For the Direct/Indirect 80 Type 2 interface, this pin is the lower byte write enable signal (WEL#). • For detailed pin functions for all other interfaces, refer to the Specification for each Host. For detailed pin functions for each host bus interface, see Section 5.4, “Host Bus Interface Pin Mapping” on page 44.
WE1#	IO	23	G5	BLNC4P1	HVDD1	Z	This input/output pin has multiple functions. • For the Generic interface, this pin is the upper byte write enable signal (WE1#). • For the Direct/Indirect 68 interface, this pin is the upper data strobe (UDS#). • For the Direct/Indirect 80 Type 1 interface, this pin is the upper byte enable signal (UBE#). • For the Direct/Indirect 80 Type 2 interface, this pin is the upper byte write enable signal (WEU#). • For detailed pin functions for all other interfaces, refer to the Specification for each Host. For detailed pin functions for each host bus interface, see Section 5.4, “Host Bus Interface Pin Mapping” on page 44. Note: For the MPC555 host interface, this pin is the burst inhibit output (BI#).
BS#	I	26	H3	ICU1	HVDD1	Z	This input pin has multiple functions. For detailed pin functions for each host bus interface, see Section 5.4, “Host Bus Interface Pin Mapping” on page 44.
BURST#	I	34	J4	ICU1	HVDD1	Z	For the MPC555 interface, the internal pull-up on this pin is disabled. If this pin is not used, it must be connected to HVDD1. For all other interfaces, the internal pull-up is enabled and this pin must be left unconnected.
BDIP#	I	35	J6	ICU1	HVDD1	Z	For the MPC555 interface, the internal pull-up on this pin is disabled. If this pin is not used, it must be connected to HVDD1. For all other interfaces, the internal pull-up is enabled and this pin must be left unconnected.

Table 5-3 : Host Interface Pin Descriptions

Pin Name	Type	QFP Pin#	PBGA Ball #	Cell	Power	RESET# State	Description
WAIT#	O	31	J2	OTLN4	HVDD1	Z	During a data transfer, this output pin is driven active to force the system to insert wait states. It is driven inactive to indicate the completion of a data transfer. WAIT# is released to the high impedance state after the data transfer is complete. The active polarity is configured according to the Host Bus Interface selected using CNF[4:0]. For detailed pin functions for each host bus interface, see Section 5.4, “Host Bus Interface Pin Mapping” on page 44.
RESET#	I	32	J3	ICS	HVDD1	1	This active low input sets all internal registers to their default state and to force all signals to their inactive states. **Note:** For details on RESET# timing, refer to Section 7.3, “RESET# Timing” on page 62 and Section 7.2, “Power Supply Sequence” on page 60.
BUSCLK	I	21	G2	ICU1	HVDD1	Z	This input clock is typically used as an external clock source for the Host CPU Bus interface. For detailed pin functions for each host bus interface, see Section 5.4, “Host Bus Interface Pin Mapping” on page 44. For details on the S1D13513 clock structure, refer to Section 9, “Clocks” on page 124.
INT1#	O	30	H5	OTLN4	HVDD1	Z	This output pin is the primary IRQ output from the S1D13513. When enabled (REG[002Ah] bit 15 = 1b), it can output all internal IRQ requests to the Host. The output and polarity is configurable using REG[002Ah].
INT2#	O	—	H6	OTLN4	HVDD1	Z	This output pin is the secondary IRQ output from the S1D13513 and can output all internal IRQ requests to the Host. Additionally, it can be configured to serve as a dedicated output for a specific Interrupt (see REG[0022h). The output and polarity is configurable using REG[002Ah]. **Note:** The QFP package does not support INT2#.

5.2.2 LCD Interface

For a pin mapping summary for each panel type, refer to Table 5-15:, “LCD Interface Pin Mapping for TFT Panels,” on page 48.

Note

The QFP package does not support 24-bit panels.

Table 5-4: LCD Interface Pin Descriptions

Pin Name	Type	QFP Pin#	PBGA Ball #	Cell	Power	RESET# State	Description
FPDAT[23:18]	IO	—	K4,R3,T3, P5,K5,R5	BLNCS4D1	HVDD2	0	These input/output pins have multiple functions. • For 24-bit panels (REG[0800h] bits 10-8 = 011b), these bits are Panel Data bits 23-18. Note that GPIOH[5:0] must also be configured for 24-bit panels by setting REG[0C1Eh] bits 11-0 to 555h. • For 18-bit panels (REG[0800h] bits 10-8 = 001b or 010b), these pins can be used as general purpose IO pins (GPIOH[5:0]). For details on configuring these pins as GPIOs, refer to Section 5.6, “GPIO Pin Mapping” on page 50. • For passive panels, these pins can be used as general purpose IO pins (GPIOH[5:0]). For details on configuring these pins as GPIOs, refer to Section 5.6, “GPIO Pin Mapping” on page 50. Note: These pins have internal pull-down resistors that are controlled using REG[0468h] bits 13-8. Note: The QFP package does not support 24-bit TFT panels.
FPDAT[17:0]	O	53-56, 59-64,67-74	N4,P4,T2, R4,T4,T5, N5,M5,P6, R6,K6,M6, L7,P7,R7, T7,N7,M7	OTLN4	HVDD2	L	These output pins are Panel Data bits 17-0. For a summary of pin usage for each panel type, see Section 5.5, “LCD Interface Pin Mapping” on page 48.
FPFRAME	O	78	R8	OTLN4	HVDD2	L	This output pin is Frame Pulse for the LCD panel.
FPLINE	O	79	T8	OTLN4	HVDD2	L	This output pin is Line Pulse for the LCD panel.
FPSHIFT	O	77	P8	OTLN4	HVDD2	L	This output pin is Shift Clock for the LCD panel.
FPDRDY	O	80	M8	OTLN4	HVDD2	L	This output pin has multiple functions. • Display Enable (DRDY) for TFT panels • LCD backplane bias signal (MOD) for passive LCD panels

Table 5-4: LCD Interface Pin Descriptions

Pin Name	Type	QFP Pin#	PBGA Ball #	Cell	Power	RESET# State	Description
GPIOG4	IO	81	N8	BLNCS4D1	HVDD2	0	This input/output pin has multiple functions. • When REG[0C1Ah] bits 9-8 = 00b, this pin is configured as a general purpose input. (default) • When REG[0C1Ah] bits 9-8 = 01b, this pin is used for HR-TFT panels (SPR). • When REG[0C1Ah] bits 9-8 = 10b, this pin is configured as a general purpose output. • When REG[0C1Ah] bits 9-8 = 11b, this pin is used for TFT panels with a serial command interface (see Table 5-15:, “LCD Interface Pin Mapping for TFT Panels,” on page 48). Note: This pin has an internal pull-down resistor that is controlled using REG[0468h] bit 4.
GPIOG3	IO	82	T9	BLNCS4D1	HVDD2	0	This input/output pin has multiple functions. • When REG[0C1Ah] bits 7-6 = 00b, this pin is configured as a general purpose input. (default) • When REG[0C1Ah] bits 7-6 = 01b, this pin is used for HR-TFT panels (SPL). • When REG[0C1Ah] bits 7-6 = 10b, this pin is configured as a general purpose output. • When REG[0C1Ah] bits 7-6 = 11b, this pin is used for TFT panels with a serial command interface (see Table 5-15:, “LCD Interface Pin Mapping for TFT Panels,” on page 48). Note: This pin has an internal pull-down resistor that is controlled using REG[0468h] bit 3.
GPIOG2	IO	83	R9	BLNCS4D1	HVDD2	0	This input/output pin has multiple functions. • When REG[0C1Ah] bits 5-4 = 00b, this pin is configured as a general purpose input. (default) • When REG[0C1Ah] bits 5-4 = 01b, this pin is used for HR-TFT panels (REV). • When REG[0C1Ah] bits 5-4 = 10b, this pin is configured as a general purpose output. • When REG[0C1Ah] bits 5-4 = 11b, this pin is used for TFT panels with a serial command interface (see Table 5-15:, “LCD Interface Pin Mapping for TFT Panels,” on page 48). Note: This pin has an internal pull-down resistor that is controlled using REG[0468h] bit 2.

Table 5-4: LCD Interface Pin Descriptions

Pin Name	Type	QFP Pin#	PBGA Ball #	Cell	Power	RESET# State	Description
GPIOG1	IO	84	P9	BLNCS4D1	HVDD2	0	This input/output pin has multiple functions. • When REG[0C1Ah] bits 3-2 = 00b, this pin is configured as a general purpose input. (default) • When REG[0C1Ah] bits 3-2 = 01b, this pin is used for HR-TFT panels (CLS). • When REG[0C1Ah] bits 3-2 = 10b, this pin is configured as a general purpose output. • When REG[0C1Ah] bits 3-2 = 11b, this pin is used for TFT panels with a serial command interface (see Table 5-15:, “LCD Interface Pin Mapping for TFT Panels,” on page 48). Note: This pin has an internal pull-down resistor that is controlled using REG[0468h] bit 1.
GPIOG0	IO	85	L9	BLNCS4D1	HVDD2	0	This input/output pin has multiple functions. • When REG[0C1Ah] bits 1-0 = 00b, this pin is configured as a general purpose input. (default) • When REG[0C1Ah] bits 1-0 = 01b, this pin is used for HR-TFT panels (PS). • When REG[0C1Ah] bits 1-0 = 10b, this pin is configured as a general purpose output. • When REG[0C1Ah] bits 1-0 = 11b, this pin is used for TFT panels with a serial command interface (see Table 5-15:, “LCD Interface Pin Mapping for TFT Panels,” on page 48). Note: This pin has an internal pull-down resistor that is controlled using REG[0468h] bit 0.

5.2.3 SDRAM Interface

Table 5-5: SDRAM Interface Pin Descriptions

Pin Name	Type	QFP Pin#	PBGA Ball #	Cell	Power	RESET# State	Description
MEMA12	O	—	F15	OTLN4	HVDD5	L	This output pin is used for SDRAM bank row/column address mapping. Note: The QFP package does not support 256Mbit or 512Mbit SDRAM sizes.
MEMA[11:0]	O	146-149, 176-183	F14,F13, F12,E16,A9, E9,F9,A8, B8,C8,E8, F8	OTLN4	HVDD5	L	These output pins are used for SDRAM bank row/column address mapping.
MEMBA[1:0]	O	144-145	G12,F16	OTLN4	HVDD5	L	These output pins are used to select the SDRAM Bank Address.
MEMCS#	O	140	H11	OTLN4	HVDD5	H	This output pin is chip select for the SDRAM.
MEMRAS#	O	141	G15	OTLN4	HVDD5	H	This output pin is the RAS# for the SDRAM.
MEMCAS#	O	142	G14	OTLN4	HVDD5	H	This output pin is the CAS# for the SDRAM.
MEMWE#	O	143	G16	OTLN4	HVDD5	H	This output pin is write enable for the SDRAM.
MEMDQ[31:16]	IO	—	D16,E13, C16,C15, A15,B14, B13,A13, B12,A11, C11,E11, D10,E10, F10,D9	BLNC4D2	HVDD5	0	These input/output pins are the upper data bus used for x32 SDRAM configurations. For x16 SDRAM configurations, these pins must be left unconnected since they have internal pull-down resistors. Note: The QFP package does not support x32 SDRAM.
MEMDQ[15:0]	IO	152-155, 158-161, 164-167, 170-173	E14,D15, D14,B16, B15,A14, C13,D12, A12,E12, D11,B11, C10,B10, A10,C9	BLNC4D2	HVDD5	0	These input/output pins are the data bus for the SDRAM. They are used for both x16 and x32 configurations. These pins have internal pull-down resistors.
MEMDQM[3:2]	O	—	J11,H15	OTLN4	HVDD5	L	These output pins are the upper byte enables used for x32 SDRAM configurations. For x16 SDRAM configurations, they should be left unconnected. Note: The QFP package does not support x32 SDRAM.
MEMDQM[1:0]	O	135-136	H14,H13	OTLN4	HVDD5	L	These output pins are the byte enables for the SDRAM. They are used for both x16 and x32 configurations.
MEMCLK	O	134	J16	OTLN8	HVDD5	H	This output pin is the clock for the SDRAM.
MEMCKE	O	137	H16	OTLN4	HVDD5	H	This output pin is the clock enable for the SDRAM.

5.2.4 GPIO / Multi Function Interface

The S1D13513 supports many features using the general purpose IO pins. All GPIO pins can be configured, using REG[0C00h] ~ REG[0C1Eh], as an input, output, Non-GPIO function #1, or Non-GPIO function #2, if available. For summary of the functionality for all GPIO pins, see Section 5.6, “GPIO Pin Mapping” on page 50.

The QFP package does not include all GPIO pins. Any features that are supported using pins unavailable on the QFP package, such as the Camera2 interface and keypad interface, are not supported.

Table 5-6: GPIO / Multi Function Pin Descriptions

Pin Name	Type	QFP Pin#	PBGA Ball #	Cell	Power	RESET# State	Description
GPIOA7	IO	87	M9	BLNCS4D1	HVDD3	Z	This pin has multiple functions. • When REG[0C02h] bits 15-14 = 00b, this pin is configured as a general purpose input. (default) • When REG[0C02h] bits 15-14 = 01b, this pin is configured as the PWM Red output (PWMR). • When REG[0C02h] bits 15-14 = 10b, this pin is configured as a general purpose output. • When REG[0C02h] bits 15-14 = 11b, this pin is configured as the Camera2 data pin 7 input (CM2DAT7). Note: The pull-down resistor on this pin is only active during test mode.
GPIOA6	IO	88	L10	BLNCS4D1	HVDD3	Z	This pin has multiple functions. • When REG[0C02h] bits 13-12 = 00b, this pin is configured as a general purpose input. (default) • When REG[0C02h] bits 13-12 = 01b, this pin is configured as the PWM Green output (PWMG). • When REG[0C02h] bits 13-12 = 10b, this pin is configured as a general purpose output. • When REG[0C02h] bits 13-12 = 11b, this pin is configured as the Camera2 data pin 6 input (CM2DAT6). Note: The pull-down resistor on this pin is only active during test mode.
GPIOA5	IO	89	M10	BLNCS4D1	HVDD3	Z	This pin has multiple functions. • When REG[0C02h] bits 11-10 = 00b, this pin is configured as a general purpose input. (default) • When REG[0C02h] bits 11-10 = 01b, this pin is configured as the PWM Blue output (PWMB). • When REG[0C02h] bits 11-10 = 10b, this pin is configured as a general purpose output. • When REG[0C02h] bits 11-10 = 11b, this pin is configured as the Camera2 data pin 5 input (CM2DAT5). Note: The pull-down resistor on this pin is only active during test mode.

Table 5-6: GPIO / Multi Function Pin Descriptions

Pin Name	Type	QFP Pin#	PBGA Ball #	Cell	Power	RESET# State	Description
GPIOA4	IO	90	T11	BLNCS4D1	HVDD3	Z	This pin has multiple functions. • When REG[0C02h] bits 9-8 = 00b, this pin is configured as a general purpose input. (default) • When REG[0C02h] bits 9-8 = 01b, this pin is configured as keypad interface input (KEYX4). • When REG[0C02h] bits 9-8 = 10b, this pin is configured as a general purpose output. • When REG[0C02h] bits 9-8 = 11b, this pin is configured as the Camera2 data pin 4 input (CM2DAT4). Note: The pull-down resistor on this pin is only active during test mode.
GPIOA3	IO	—	T10	BLNCS4D1	HVDD3	Z	This pin has multiple functions. • When REG[0C02h] bits 7-6 = 00b, this pin is configured as a general purpose input. (default) • When REG[0C02h] bits 7-6 = 01b, this pin is configured as keypad interface input (KEYX3). • When REG[0C02h] bits 7-6 = 10b, this pin is configured as a general purpose output. • When REG[0C02h] bits 7-6 = 11b, this pin is configured as the Camera2 data pin 3 input (CM2DAT3). Note: The pull-down resistor on this pin is only active during test mode.
GPIOA2	IO	—	R10	BLNCS4D1	HVDD3	Z	This pin has multiple functions. • When REG[0C02h] bits 5-4 = 00b, this pin is configured as a general purpose input. (default) • When REG[0C02h] bits 5-4 = 01b, this pin is configured as keypad interface input (KEYX2). • When REG[0C02h] bits 5-4 = 10b, this pin is configured as a general purpose output. • When REG[0C02h] bits 5-4 = 11b, this pin is configured as the Camera2 data pin 2 input (CM2DAT2). Note: The pull-down resistor on this pin is only active during test mode.
GPIOA1	IO	—	N10	BLNCS4D1	HVDD3	Z	This pin has multiple functions. • When REG[0C02h] bits 3-2 = 00b, this pin is configured as a general purpose input. (default) • When REG[0C02h] bits 3-2 = 01b, this pin is configured as keypad interface input (KEYX1). • When REG[0C02h] bits 3-2 = 10b, this pin is configured as a general purpose output. • When REG[0C02h] bits 3-2 = 11b, this pin is configured as the Camera2 data pin 1 input (CM2DAT1). Note: The pull-down resistor on this pin is only active during test mode.

Table 5-6: GPIO / Multi Function Pin Descriptions

Pin Name	Type	QFP Pin#	PBGA Ball #	Cell	Power	RESET# State	Description
GPIOA0	IO	—	P10	BLNCS4D1	HVDD3	Z	This pin has multiple functions. • When REG[0C02h] bits 1-0 = 00b, this pin is configured as a general purpose input. (default) • When REG[0C02h] bits 1-0 = 01b, this pin is configured as keypad interface input (KEYX0). • When REG[0C02h] bits 1-0 = 10b, this pin is configured as a general purpose output. • When REG[0C02h] bits 1-0 = 11b, this pin is configured as the Camera2 data pin 0 input (CM2DAT0). Note: The pull-down resistor on this pin is only active during test mode.
GPIOB7	IO	93	M11	BLNCS4D1	HVDD3	Z	This pin has multiple functions. • When REG[0C06h] bits 15-14 = 00b, this pin is configured as a general purpose input. (default) • When REG[0C06h] bits 15-14 = 01b, this pin is configured as the PWM White output (PWMW). • When REG[0C06h] bits 15-14 = 10b, this pin is configured as a general purpose output. • When REG[0C06h] bits 15-14 = 11b, this pin is configured as the camera strobe output (CMSTROUT). Note: The pull-down resistor on this pin is only active during test mode.
GPIOB6	IO	94	N11	BLNCS4D1	HVDD3	Z	This pin has multiple functions. • When REG[0C06h] bits 13-12 = 00b, this pin is configured as a general purpose input. (default) • When REG[0C06h] bits 13-12 = 01b, this pin is SDA for the I2C interface (Pseudo Open Drain Pin). • When REG[0C06h] bits 13-12 = 10b, this pin is configured as a general purpose output. • When REG[0C06h] bits 13-12 = 11b, this pin is reserved. Note: The pull-down resistor on this pin is only active during test mode.
GPIOB5	IO	95	T13	BLNCS4D1	HVDD3	Z	This pin has multiple functions. • When REG[0C06h] bits 111-10 = 00b, this pin is configured as a general purpose input. (default) • When REG[0C06h] bits 11-10 = 01b, this pin is SCL for the I2C interface (Pseudo Open Drain Pin). • When REG[0C06h] bits 11-10 = 10b, this pin is configured as a general purpose output. • When REG[0C06h] bits 11-10 = 11b, this pin is reserved. Note: The pull-down resistor on this pin is only active during test mode.

Table 5-6: GPIO / Multi Function Pin Descriptions

Pin Name	Type	QFP Pin#	PBGA Ball #	Cell	Power	RESET# State	Description
GPIOB4	IO	96	T14	BLNCS4D1	HVDD3	Z	This pin has multiple functions. • When REG[0C06h] bits 9-8 = 00b, this pin is configured as a general purpose input. (default) • When REG[0C06h] bits 9-8 = 01b, this pin is configured as keypad interface input (KEYY4). • When REG[0C06h] bits 9-8 = 10b, this pin is configured as a general purpose output. • When REG[0C06h] bits 9-8 = 11b, this pin is reserved. Note: The pull-down resistor on this pin is only active during test mode.
GPIOB3	IO	—	T12	BLNCS4D1	HVDD3	Z	This pin has multiple functions. • When REG[0C06h] bits 7-6 = 00b, this pin is configured as a general purpose input. (default) • When REG[0C06h] bits 7-6 = 01b, this pin is configured as keypad interface input (KEYY3). • When REG[0C06h] bits 7-6 = 10b, this pin is configured as a general purpose output. • When REG[0C06h] bits 7-6 = 11b, this pin is configured as the vertical sync input for the Camera2 interface (CM2VREF). Note: The pull-down resistor on this pin is only active during test mode.
GPIOB2	IO	—	R12	BLNCS4D1	HVDD3	Z	This pin has multiple functions. • When REG[0C06h] bits 5-4 = 00b, this pin is configured as a general purpose input. (default) • When REG[0C06h] bits 5-4 = 01b, this pin is configured as keypad interface input (KEYY2). • When REG[0C06h] bits 5-4 = 10b, this pin is configured as a general purpose output. • When REG[0C06h] bits 5-4 = 11b, this pin is configured as the horizontal sync input for the Camera2 interface (CM2HREF). Note: The pull-down resistor on this pin is only active during test mode.
GPIOB1	IO	—	P11	BLNCS4D1	HVDD3	Z	This pin has multiple functions. • When REG[0C06h] bits 3-2 = 00b, this pin is configured as a general purpose input. (default) • When REG[0C06h] bits 3-2 = 01b, this pin is configured as keypad interface input (KEYY1). • When REG[0C06h] bits 3-2 = 10b, this pin is configured as a general purpose output. • When REG[0C06h] bits 3-2 = 11b, this pin is configured as the master clock output for the Camera2 interface (CM2CLKOUT). Note: The pull-down resistor on this pin is only active during test mode.

Table 5-6: GPIO / Multi Function Pin Descriptions

Pin Name	Type	QFP Pin#	PBGA Ball #	Cell	Power	RESET# State	Description
GPIOB0	IO	—	R13	BLNCS4D1	HVDD3	Z	This pin has multiple functions. • When REG[0C06h] bits 1-0 = 00b, this pin is configured as a general purpose input. (default) • When REG[0C06h] bits 1-0 = 01b, this pin is configured as keypad interface input (KEYY0). • When REG[0C06h] bits 1-0 = 10b, this pin is configured as a general purpose output. • When REG[0C06h] bits 1-0 = 11b, this pin is configured as the camera clock input for the Camera2 interface (CM2CLKIN). Note: The pull-down resistor on this pin is only active during test mode.
GPIOC[7:0]	IO	107-114	R16,P15, N14,M14, N15,P16, M12,N16	BLNCS4D1	HVDD4	0	These pins have multiple functions. • When the appropriate bits from REG[0C0Ah] are set to 00b, the corresponding pins are configured as general purpose inputs. (default) • When the appropriate bits from REG[0C0Ah] are set to 01b, the corresponding pins are configured as YUV Digital Output pins (YUVDAT[7:0] for the external video encoder. • When the appropriate bits from REG[0C0Ah] are set to 10b, the corresponding pins are configured as general purpose outputs. • When the appropriate bits from REG[0C0Ah] are set to 11b, the corresponding pins are configured as Camera1 data pin (CM1DAT[7:0]). Note: These pins have internal pull-down resistors that are controlled using REG[0464h] bits 7-0.
GPIOD3	IO	115	M15	BLNCS4D1	HVDD4	0	This pin has multiple functions. • When REG[0C0Eh] bits 7-6 = 00b, this pin is configured as a general purpose input. (default) • When REG[0C0Eh] bits 7-6 = 01b, this pin is configured as the digital audio input for the PWM interface (AUDIN). • When REG[0C0Eh] bits 7-6 = 10b, this pin is configured as a general purpose output. • When REG[0C0Eh] bits 7-6 = 11b, this pin is configured as the camera clock input for the Camera1 interface (CM1CLKIN). Note: This pin has an internal pull-down resistor that is controlled using REG[0464h] bit 11.

Table 5-6: GPIO / Multi Function Pin Descriptions

Pin Name	Type	QFP Pin#	PBGA Ball #	Cell	Power	RESET# State	Description
GPIOD2	IO	116	L14	BLNCS4D1	HVDD4	0	This pin has multiple functions. • When REG[0C0Eh] bits 5-4 = 00b, this pin is configured as a general purpose input. (default) • When REG[0C0Eh] bits 5-4 = 01b, this pin is configured as the clock output for the external video encoder (YUVCLKO). • When REG[0C0Eh] bits 5-4 = 10b, this pin is configured as a general purpose output. • When REG[0C0Eh] bits 5-4 = 11b, this pin is configured as the master clock output for the Camera1 interface (CM1CLKOUT). Note: This pin has an internal pull-down resistor that is controlled using REG[0464h] bit 10.
GPIOD1	IO	117	L13	BLNCS4D1	HVDD4	0	This pin has multiple functions. • When REG[0C0Eh] bits 3-2 = 00b, this pin is configured as a general purpose input. (default) • When REG[0C0Eh] bits 3-2 = 01b, this pin is configured as the horizontal sync for the external video encoder (YUVHSYNC). • When REG[0C0Eh] bits 3-2 = 10b, this pin is configured as a general purpose output. • When REG[0C0Eh] bits 3-2 = 11b, this pin is configured as the horizontal sync input for the Camera1 interface (CM1HREF). Note: This pin has an internal pull-down resistor that is controlled using REG[0464h] bit 9.
GPIOD0	IO	118	M16	BLNCS4D1	HVDD4	0	This pin has multiple functions. • When REG[0C0Eh] bits 1-0 = 00b, this pin is configured as a general purpose input. (default) • When REG[0C0Eh] bits 1-0 = 01b, this pin is configured as the vertical sync for the external video encoder (YUVVSYNC). • When REG[0C0Eh] bits 1-0 = 10b, this pin is configured as a general purpose output. • When REG[0C0Eh] bits 1-0 = 11b, this pin is configured as the vertical sync input for the Camera1 interface (CM1VREF). Note: This pin has an internal pull-down resistor that is controlled using REG[0464h] bit 8.

5.2.5 Miscellaneous

Table 5-7: Miscellaneous Pin Descriptions

Pin Name	Type	QFP Pin#	PBGA Ball #	Cell	Power	RESET# State	Description
CNF[8:0]	I	122-130	K14,L16, K15,K12, K16,J14, K11,J13,J15	ICD2	HVDD4	0	These inputs are used to configure the S1D13513 and must be connected to IOVDD or VSS. The states of these pins are latched at RESET#. These pins have internal pull-down resistors which can be disabled by software after reset (see REG[046Eh]). For more information, see Section 5.3, “Summary of Configuration Options” on page 41.
OSCI1	I	40	L1	ILTR	OSCVDD1	0/1	Crystal input. If an external Oscillator circuit or clock generator is used, connect it to this pin. For details on the clock structure, see Section 9, “Clocks” on page 124.
OSCO1	O	41	L2	OLTR	OSCVDD1	H	Crystal output. If an external Oscillator is used, this pin should be left unconnected. For details on the clock structure, see Section 9, “Clocks” on page 124.
OSCI2	I	45	M1	ILTR	OSCVDD2	0/1	Crystal input. If an external Oscillator circuit or clock generator is used, connect it to this pin. Typically, this input is used to provide 27MHz for the Video Encoder. For details on the clock structure, see Section 9, “Clocks” on page 124.
OSCO2	O	44	M2	OLTR	OSCVDD2	H	Crystal output. If an external Oscillator is used, this pin should be left unconnected. For details on the clock structure, see Section 9, “Clocks” on page 124.
CLKI3	I	36	K1	IC	HVDD1	0/1	Clock input. Typically, this input is used for the SDRAM clock through PLL1.
TESTEN	I	121	L15	ICSD1	HVDD4	0	This input pin is for production test only and should be left unconnected for normal operation.
VCP1	O	48	N1	OLTR	PLLVDD1	Z	This output is for production test only and should be left unconnected for normal operation.
VCP2	O	51	R1	OLTR	PLLVDD2	Z	This output is for production test only and should be left unconnected for normal operation.
TCK	I	100	P12	ICSP1	HVDD4	1	JTAG Interface pin for Boundary Scan test.
TMS	I	101	R14	ICSP1	HVDD4	1	JTAG Interface pin for Boundary Scan test.
TDI	I	102	N12	ICSP1	HVDD4	1	JTAG Interface pin for Boundary Scan test.
TDO	O	103	R15	OTLN4	HVDD4	L	JTAG Interface pin for Boundary Scan test
TRST	I	104	P13	ICSP2	HVDD4	1	JTAG Interface pin for Boundary Scan test. For normal operations, this pin must be tied to VSS or connected to RESET#.

5.2.6 Power And Ground

Table 5-8: Power And Ground Pin Descriptions

Pin Name	Type	QFP Pin#	PBGA Ball#	Cell	RESET# State	Description
COREVDD	P	29,37,86,97, 131,156, 157,208	C3,C14,F6, F11,K3,L6, L11,P3,P14	P	—	Core VDD
HVDD1	P	9,19,196	D5,E2,G4,	P	—	IOVDD for HOST interface
HVDD2	P	57,65,75	L5,L8,T6	P	—	IOVDD for LCD Panel interface
HVDD3	P	91	R11	P	—	IOVDD for Camera1
HVDD4	P	98,105,119	L12,M13, T15	P	—	IOVDD for Camera2, GPIO, etc.
HVDD5	P	132,138, 150,162, 168,174	B9,C12,E15, G11,H12, J12	P	—	IOVDD for SDRAM interface
VSS	P	10,20,38,58, 66,76,92,99, 106,120, 133,139, 151,163, 169,175, 184,197	A1,A16,D4, D8,D13,G7-G10,G13,H7-H10,J1,J7-J10,K2,K7-K10,K13,N3, N6,N9,N13, T1,T16	P	—	Common Ground
OSCVDD1	P	39	L4	P	—	VDD for OSC1
OSCVSS1	P	42	L3	P	—	GND for OSC1
OSCVDD2	P	46	M3	P	—	VDD for OSC2
OSCVSS2	P	43	M4	P	—	GND for OSC2
PLLVDD1	P	49	P1	P	—	Analog VDD for PLL1
PLLVSS1	P	47	N2	P	—	Analog GND for PLL1
PLLVDD2	P	50	P2	P	—	Analog VDD for PLL2
PLLVSS2	P	52	R2	P	—	Analog GND for PLL2

5.3 Summary of Configuration Options

These pins are used for configuration of the S1D13513 and must be connected using an external pull-up resistor (1) or left open (0) where an internal pull-down resistor is used. The state of CNF[8:0] has an effect only at the rising edge of RESET#. Changing state at any other time has no effect. In order to stop the constant current when an external pull-up resistor is used, disable the corresponding pull-down resistor (see REG[046Eh]).

Note

The CNF pins select the host bus interface and determine whether the interface is Little or Big Endian.

When a Big Endian host interface is selected, the registers must be accessed using a method which "byte-swaps" the upper and lower data byte in each register. For details on this requirement, see Section 20.6, "Register Accesses for Big Endian Host Interfaces" on page 452.

Table 5-9: Summary of Power-On/Reset Options for CNF6=0

S1D13513 Configuration Input	**Power-On/Reset State**	
	1 (connected to IO V_{DD})	**0 (connected to V_{SS})**
CNF[8:7]	Selects the PLL1 Clock Source as follows: **CNF8 CNF7 PLL1 Clock Source** 0 0 CLKI3 Pin 0 1 BUSCLK Pin 1 0 OSC1 Clock 1 1 OSC2 Clock	
CNF6	See Table 5-10:	See Table 5-9:
CNF5	Indirect Access	Direct Access
CNF[4:0]	Select host bus interface as follows: **CNF4 CNF3 CNF2 CNF1 CNF0 Host Bus** 0 0 0 0 0 Generic Little Endian: Active Low WAIT# with tri-state 0 0 0 0 1 Generic Little Endian: Active Low WAIT# always driven 0 0 0 1 0 Generic Little Endian: Active HIgh WAIT# with tri-state 0 0 0 1 1 Reserved 0 0 1 0 0 Generic Big Endian: Active Low WAIT# with tri-state 0 0 1 0 1 Generic Big Endian: Active Low WAIT# always driven 0 0 1 1 0 Generic Big Endian: Active HIgh WAIT# with tri-state 0 0 1 1 1 Reserved 0 1 0 0 0 MIPS/ISA Little Endian: Active Low WAIT# with tri-state 0 1 0 0 1 MIPS/ISA Little Endian: Active Low WAIT# always driven 0 1 0 1 0 MIPS/ISA Little Endian: Active HIgh WAIT# with tri-state 0 1 0 1 1 Reserved 0 1 1 0 0 MC68000 Big Endian: Active High WAIT# with tri-state 0 1 1 0 1 Reserved 0 1 1 1 0 MC68030 Big Endian: Active High WAIT# with tri-state 0 1 1 1 1 Reserved 1 0 0 0 0 PR31500/31700/TX3912 Little Endian: Active Low WAIT# with tri-state (16-bit memory accesses only) 1 0 0 0 1 PR31500/31700/TX3912 Little Endian: Active Low WAIT# always driven (16-bit memory accesses only) 1 0 0 1 0 Reserved 1 0 0 1 1 Reserved 1 0 1 0 0 Reserved 1 0 1 0 1 Reserved 1 0 1 1 0 MPC555/ Big Endian: Active High WAIT# with tri-state 1 0 1 1 1 Reserved 1 1 0 0 0 SH3 Little Endian: Active Low WAIT# with tri-state 1 1 0 0 1 SH3 Little Endian: Active Low WAIT# always driven 1 1 0 1 0 SH4 Little Endian: Active High WAIT# with tri-state 1 1 0 1 1 Reserved 1 1 1 0 0 SH3 Big Endian: Active Low WAIT# with tri-state 1 1 1 0 1 SH3 Big Endian: Active Low WAIT# always driven 1 1 1 1 0 SH4 Big Endian: Active High WAIT# with tri-state 1 1 1 1 1 Reserved	

Note

Big Endian modes are not supported for any indirect interface.

Table 5-10: Summary of Power-On/Reset Options for CNF6=1

<table>
<tr><th rowspan="2">S1D13513 Configuration Input</th><th colspan="2">Power-On/Reset State</th></tr>
<tr><th>1 (connected to IO V_{DD})</th><th>0 (connected to V_{SS})</th></tr>
<tr><td>CNF[8:7]</td><td colspan="2">Selects the PLL1 Clock Source as follows:
<table>
<tr><th>CNF8</th><th>CNF7</th><th>PLL1 Clock Source</th></tr>
<tr><td>0</td><td>0</td><td>CLKI3 Pin</td></tr>
<tr><td>0</td><td>1</td><td>BUSCLK Pin</td></tr>
<tr><td>1</td><td>0</td><td>OSC1 Clock</td></tr>
<tr><td>1</td><td>1</td><td>OSC2 Clock</td></tr>
</table></td></tr>
<tr><td>CNF6</td><td>See Table 5-10:</td><td>See Table 5-9:</td></tr>
<tr><td>CNF5</td><td>Big Endian (see Note 1)</td><td>Little Endian</td></tr>
<tr><td>CNF[4:0]</td><td colspan="2">Select host bus interface as follows:
<table>
<tr><th>CNF4</th><th>CNF3</th><th>CNF2</th><th>CNF1</th><th>CNF0</th><th>Host Bus</th></tr>
<tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>Parallel Direct 80 Type 2: 1 CS# (see Note 2)</td></tr>
<tr><td>0</td><td>0</td><td>0</td><td>0</td><td>1</td><td>Reserved</td></tr>
<tr><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>Parallel Indirect 80 Type 2</td></tr>
<tr><td>0</td><td>0</td><td>0</td><td>1</td><td>1</td><td>Reserved</td></tr>
<tr><td>0</td><td>0</td><td>1</td><td>0</td><td>0</td><td>Parallel Direct 80 Type 1: 1 CS#</td></tr>
<tr><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td><td>Parallel Direct 68: 1 CS#</td></tr>
<tr><td>0</td><td>0</td><td>1</td><td>1</td><td>0</td><td>Parallel Indirect 80 Type 1</td></tr>
<tr><td>0</td><td>0</td><td>1</td><td>1</td><td>1</td><td>Parallel Indirect 68</td></tr>
<tr><td>0</td><td>1</td><td>0</td><td>0</td><td>0</td><td>Parallel Direct 80 Type 2: 2 CS# (see Note 2)</td></tr>
<tr><td>0</td><td>1</td><td>0</td><td>0</td><td>1</td><td>Reserved</td></tr>
<tr><td>0</td><td>1</td><td>0</td><td>1</td><td>0</td><td>Reserved</td></tr>
<tr><td>0</td><td>1</td><td>0</td><td>1</td><td>1</td><td>Reserved</td></tr>
<tr><td>0</td><td>1</td><td>1</td><td>0</td><td>0</td><td>Parallel Direct 80 Type 1: 2 CS#</td></tr>
<tr><td>0</td><td>1</td><td>1</td><td>0</td><td>1</td><td>Parallel Direct 68: 2 CS#</td></tr>
<tr><td>0</td><td>1</td><td>1</td><td>1</td><td>0</td><td>Reserved</td></tr>
<tr><td>0</td><td>1</td><td>1</td><td>1</td><td>1</td><td>Reserved</td></tr>
<tr><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td><td>Serial on HVDD1: Data valid on falling edge</td></tr>
<tr><td>1</td><td>0</td><td>0</td><td>0</td><td>1</td><td>Serial on HVDD2: Data valid on falling edge</td></tr>
<tr><td>1</td><td>0</td><td>0</td><td>1</td><td>0</td><td>Reserved</td></tr>
<tr><td>1</td><td>0</td><td>0</td><td>1</td><td>1</td><td>Reserved</td></tr>
<tr><td>1</td><td>0</td><td>1</td><td>0</td><td>0</td><td>Reserved</td></tr>
<tr><td>1</td><td>0</td><td>1</td><td>0</td><td>1</td><td>Reserved</td></tr>
<tr><td>1</td><td>0</td><td>1</td><td>1</td><td>0</td><td>Reserved</td></tr>
<tr><td>1</td><td>0</td><td>1</td><td>1</td><td>1</td><td>Reserved</td></tr>
<tr><td>1</td><td>1</td><td>0</td><td>0</td><td>0</td><td>Serial on HVDD1: Data valid on rising edge</td></tr>
<tr><td>1</td><td>1</td><td>0</td><td>0</td><td>1</td><td>Serial on HVDD2: Data valid on rising edge</td></tr>
<tr><td>1</td><td>1</td><td>0</td><td>1</td><td>0</td><td>Reserved</td></tr>
<tr><td>1</td><td>1</td><td>0</td><td>1</td><td>1</td><td>Reserved</td></tr>
<tr><td>1</td><td>1</td><td>1</td><td>0</td><td>0</td><td>Reserved</td></tr>
<tr><td>1</td><td>1</td><td>1</td><td>0</td><td>1</td><td>Reserved</td></tr>
<tr><td>1</td><td>1</td><td>1</td><td>1</td><td>0</td><td>Reserved</td></tr>
<tr><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td><td>Reserved</td></tr>
</table></td></tr>
</table>

Note

1. When Big Endian mode is selected (CNF5=1), only memory accesses are byte swapped. Register accesses are unaffected.
2. Big Endian mode (CNF5=1) is not supported for Indirect 80, Indirect 68, and Serial interfaces.

5.4 Host Bus Interface Pin Mapping

Table 5-11: Host Bus Interface Pin Mapping 1 (Direct Interfaces)

S1D13513 Pin Name	**Generic**	**Renesas SH3/SH4**	**MIPS/ISA**	**FreeScale MC68000**	**FreeScale MC68030**	**Freescale MPC555**	**Philips PR31500 PR31700[3]**	**Toshiba TX3912[3]**
AB20	A20	A20	LatchA20	A20	A20	A11	ALE	ALE
AB19	A19	A19	SA19	A19	A19	A12	/CARDREG	CARDREG*
AB18	A18	A18	SA18	A18	A18	A13	/CARDIORD	CARDIORD*
AB17	A17	A17	SA17	A17	A17	A14	/CARDIOWR	CARDIOWR*
AB[16:13]	A[16:13]	A[16:13]	SA[16:13]	A[16:13]	A[16:13]	A[15:18]	Connected to VDD	Connected to VDD
AB[12:1]	A[12:1]	A[12:1]	SA[12:1]	A[12:1]	A[12:1]	A[19:30]	A[12:1]	A[12:1]
AB0	Connected to VSS[1]	Connected to VSS[1]	SA0	LDS#	A0	A31	A0	A0
DB[15:8]	D[15:8]	D[15:8]	SD[15:8]	D[15:8]	D[31:24]	D[0:7]	D[23:16]	D[23:16]
DB[7:0]	D[7:0]	D[7:0]	SD[7:0]	D[7:0]	D[23:16]	D[8:15]	D[31:24]	D[31:24]
WE0#	WE0#	WE0#	MEMW#	Connected to VDD	SIZ0	TSIZ1	/WE	WE*
WE1#	WE1#	WE1#	SBHE#	UDS#	DS#	BI#	/CARDxCSH	CARDxCSH*
M/R#	External Decode	External Decode	External Decode	External Decode	External Decode	External Decode	Connected to VDD	Connected to VDD
CS#	External Decode	External Decode	External Decode	External Decode	External Decode	CSn#	Connected to VDD	Connected to VDD
BS#	Connected to VDD	BS#	Connected to VDD	AS#	AS#	TS#	Connected to VDD	Connected to VDD
RD/WR#	RD1#	RD/WR#	Connected to VDD	R/W#	R/W#	RD/WR#	/CARDxCSL	CARDxCSL*
RD#	RD0#	RD#	MEMR#	Connected to VDD	SIZ1	TSIZ0	/RD	RD*
BUSCLK	BCLK	CKIO	CLK	CLK	CLK	CLKOUT	DCLKOUT	DCLKOUT
INT1#	INT pin at CPU	INT pin at CPU	INT pin at CPU	INT pin at CPU	INT pin at CPU	INT pin at CPU	INT pin at CPU	INT pin at CPU
INT2#	INT pin at CPU	INT pin at CPU	INT pin at CPU	INT pin at CPU	INT pin at CPU	INT pin at CPU	INT pin at CPU	INT pin at CPU
WAIT#	WAIT#	WAIT#/ RDY#	IOCHRDY	DTACK#	DSACK1#	TA#	/CARDxWAIT	CARDxWAIT*
RESET#	RESET#	RESET#	Inverted RESET	RESET#	RESET#	RESET#	RESET#	PON*
BURST#	Unconnected	Unconnected	Unconnected	Unconnected	Unconnected	BURST#[4]	Unconnected	Unconnected
BDIP#	Unconnected	Unconnected	Unconnected	Unconnected	Unconnected	BDIP#[4]	Unconnected	Unconnected
FPDAT18[2]	GPIOH0	GPIOH0	GPIOH0	GPIOH0	GPIOH0	GPIOH0	GPIOH0	GPIOH0
FPDAT19[2]	GPIOH1	GPIOH1	GPIOH1	GPIOH1	GPIOH1	GPIOH1	GPIOH1	GPIOH1
FPDAT20[2]	GPIOH2	GPIOH2	GPIOH2	GPIOH2	GPIOH2	GPIOH2	GPIOH2	GPIOH2
FPDAT21[2]	GPIOH3	GPIOH3	GPIOH3	GPIOH3	GPIOH3	GPIOH3	GPIOH3	GPIOH3
FPDAT22[2]	GPIOH4	GPIOH4	GPIOH4	GPIOH4	GPIOH4	GPIOH4	GPIOH4	GPIOH4
FPDAT23[2]	GPIOH5	GPIOH5	GPIOH5	GPIOH5	GPIOH5	GPIOH5	GPIOH5	GPIOH5

1. AB0 is not used in the S1D13513 internal circuitry. In this mode, it must be connected to either VSS or VDD.
2. FPDAT[23:18] are not available on the QFP package of the S1D13513.
3. Only 16-bit memory accesses are supported.
4. If the BURST# and BDIP# pins are not connected to the host CPU, they must be connected to VDD.

Table 5-12: Host Bus Interface Pin Mapping 2 (Indirect Interfaces)

S1D13513 Pin Name	**Generic**	**Renesas SH3/SH4**	**MIPS/ISA**	**FreeScale MC68000**	**FreeScale MC68030**	**Freescale MPC555**	**Philips PR31500 PR31700[3]**	**Toshiba TX3912[3]**
AB20	Unconnected	Unconnected	Unconnected	Unconnected	Unconnected	Unconnected	ALE	ALE
AB19	Unconnected	Unconnected	Unconnected	Unconnected	Unconnected	Unconnected	/CARDREG	CARDREG*
AB18	Unconnected	Unconnected	Unconnected	Unconnected	Unconnected	Unconnected	/CARDIORD	CARDIORD*
AB17	Unconnected	Unconnected	Unconnected	Unconnected	Unconnected	Unconnected	/CARDIOWR	CARDIOWR*
AB[16:13]	Unconnected	Unconnected	Unconnected	Unconnected	Unconnected	Unconnected	Connected to VDD	Connected to VDD
AB[12:3]	Unconnected	Unconnected	Unconnected	Unconnected	Unconnected	Unconnected	Unconnected	Unconnected
AB[2:1]	A[2:1]	A[2:1]	SA[2:1]	A[2:1]	A[2:1]	A[29:30]	A[2:1	A[2:1
AB0	Connected to VSS[1]	Connected to VSS[1]	Connected to VSS	LDS#	A0	Connected to VSS	Connected to VSS	Connected to VSS
DB[15:8]	D[15:8]	D[15:8]	SD[15:8]	D[15:8]	D[31:24]	D[0:7]	D[23:16]	D[23:16]
DB[7:0]	D[7:0]	D[7:0]	SD[7:0]	D[7:0]	D[23:16]	D[8:15]	D[31:24]	D[31:24]
WE0#	WE0#	WE0#	MEMW#	Connected to VDD	SIZ0	TSIZ1	/WE	WE*
WE1#	WE1#	WE1#	SBHE#	UDS#	DS#	BI#	/CARDxCSH	CARDxCSH*
M/R#	External Decode	External Decode	External Decode	External Decode	External Decode	External Decode	Connected to VDD	Connected to VDD
CS#	External Decode	External Decode	External Decode	External Decode	External Decode	CSn#	Connected to VDD	Connected to VDD
BS#	Connected to VDD	BS#	Connected to VDD	AS#	AS#	TS#	Connected to VDD	Connected to VDD
RD/WR#	RD1#	RD/WR#	Connected to VDD	R/W#	R/W#	RD/WR#	/CARDxCSL	CARDxCSL*
RD#	RD0#	RD#	MEMR#	Connected to VDD	SIZ1	TSIZ0	/RD	RD*
BUSCLK	BCLK	CKIO	CLK	CLK	CLK	CLKOUT	DCLKOUT	DCLKOUT
INT1#	INT pin at CPU	INT pin at CPU	INT pin at CPU	INT pin at CPU	INT pin at CPU	INT pin at CPU	INT pin at CPU	INT pin at CPU
INT2#	INT pin at CPU	INT pin at CPU	INT pin at CPU	INT pin at CPU	INT pin at CPU	INT pin at CPU	INT pin at CPU	INT pin at CPU
WAIT#	WAIT#	WAIT#/ RDY#	IOCHRDY	DTACK#	DSACK1#	TA#	/CARDxWAIT	CARDxWAIT*
RESET#	RESET#	RESET#	Inverted RESET	RESET#	RESET#	RESET#	RESET#	PON*
BURST#	Unconnected	Unconnected	Unconnected	Unconnected	Unconnected	BURST#[4]	Unconnected	Unconnected
BDIP#	Unconnected	Unconnected	Unconnected	Unconnected	Unconnected	BDIP#[4]	Unconnected	Unconnected
FPDAT18[2]	GPIOH0	GPIOH0	GPIOH0	GPIOH0	GPIOH0	GPIOH0	GPIOH0	GPIOH0
FPDAT19[2]	GPIOH1	GPIOH1	GPIOH1	GPIOH1	GPIOH1	GPIOH1	GPIOH1	GPIOH1
FPDAT20[2]	GPIOH2	GPIOH2	GPIOH2	GPIOH2	GPIOH2	GPIOH2	GPIOH2	GPIOH2
FPDAT21[2]	GPIOH3	GPIOH3	GPIOH3	GPIOH3	GPIOH3	GPIOH3	GPIOH3	GPIOH3
FPDAT22[2]	GPIOH4	GPIOH4	GPIOH4	GPIOH4	GPIOH4	GPIOH4	GPIOH4	GPIOH4
FPDAT23[2]	GPIOH5	GPIOH5	GPIOH5	GPIOH5	GPIOH5	GPIOH5	GPIOH5	GPIOH5

1. AB0 is not used in the S1D13513 internal circuitry. In this mode, it must be connected to either VSS or VDD.
2. FPDAT[23:18] are not available on the QFP package of the S1D13513.
3. Only 16-bit memory accesses are supported.
4. If the BURST# and BDIP# pins are not connected to the host CPU, they must be connected to VDD.

Table 5-13: Host Bus Interface Pin Mapping 3 (Direct Interfaces)

S1D13513 Pin Name	**Direct 68**	**Direct 80 Type 1**	**Direct 80 Type 2**
AB[20:1]	A[20:1]	A[20:1]	A[20:1]
AB0	Connected to VSS	Connected to VSS	Connected to VSS
DB[15:0]	D[15:0]	D[15:0]	D[15:0]
WE0#	LDS#	LBE#	WEL#
WE1#	UDS#	UBE#	WEU#
M/R#	Address (1CS#), chip/selection	Address (1CS#), chip/selection	Address (1CS#), chip/selection
CS#	CS#	CS#	CS#
BS#	Connected to VDD	Connected to VDD	Connected to VDD
RD/WR#	R/W#	WE#	HVDD1
RD#	HVDD1	RD#	RD#
BUSCLK	Connected to VSS	Connected to VSS	Connected to VSS
INT1#	INT pin at CPU	INT pin at CPU	INT pin at CPU
INT2#	INT pin at CPU	INT pin at CPU	INT pin at CPU
WAIT#	WAIT#	WAIT#	WAIT#
RESET#	RESET#	RESET#	RESET#
BURST#	Unconnected	Unconnected	Unconnected
BDIP#	Unconnected	Unconnected	Unconnected
FPDAT18[1]	GPIOH0	GPIOH0	GPIOH0
FPDAT19[1]	GPIOH1	GPIOH1	GPIOH1
FPDAT20[1]	GPIOH2	GPIOH2	GPIOH2
FPDAT21[1]	GPIOH3	GPIOH3	GPIOH3
FPDAT22[1]	GPIOH4	GPIOH4	GPIOH4
FPDAT23[1]	GPIOH5	GPIOH5	GPIOH5

1. FPDAT[23:18] are not available on the QFP package of the S1D13513.

Table 5-14: Host Bus Interface Pin Mapping 4 (Indirect Interfaces)

S1D13513 Pin Name	Indirect 68	Indirect 80 Type 1	Indirect 80 Type 2	Serial on HVDD1	Serial on HVDD2
AB[20:3]	Unconnected	Unconnected	Unconnected	Unconnected	Unconnected
AB2	A2	A2	A2	Unconnected	Unconnected
AB1	A1	A1	A1	Unconnected	Unconnected
AB0	Connected to VSS	Connected to VSS	Connected to VSS	SA0	Connected to VSS
DB[15:2	D[15:2]	D[15:2]	D[15:2]	Unconnected	Unconnected
DB1	D1	D1	D1	SO	Unconnected
DB0	D0	D0	D0	SI	Connected to VSS
WE0#	LDS#	LBE#	WEL#	Unconnected	Unconnected
WE1#	UDS#	UBE#	WEU#	Unconnected	Unconnected
M/R#	Unconnected	Unconnected	Unconnected	Unconnected	Unconnected
CS#	CS#	CS#	CS#	SCS#	Connected to VSS
BS#	Connected to VDD	Connected to VDD	Connected to VDD	Unconnected	Unconnected
RD/WR#	R/W#	WE#	HVDD1	Unconnected	Unconnected
RD#	HVDD1	RD#	RD#	Unconnected	Unconnected
BUSCLK	Connected to VSS	Connected to VSS	Connected to VSS	SCLK	Connected to VSS
INT1#	INT pin at CPU	INT pin at CPU	INT pin at CPU	INT pin at CPU	INT pin at CPU
INT2#	INT pin at CPU	INT pin at CPU	INT pin at CPU	INT pin at CPU	INT pin at CPU
WAIT#	WAIT#	WAIT#	WAIT#	Unconnected	Unconnected
RESET#	RESET#	RESET#	RESET#	RESET#	RESET#
BURST#	Unconnected	Unconnected	Unconnected	Unconnected	Unconnected
BDIP#	Unconnected	Unconnected	Unconnected	Unconnected	Unconnected
FPDAT18[1]	GPIOH0	GPIOH0	GPIOH0	GPIOH0	SCS#
FPDAT19[1]	GPIOH1	GPIOH1	GPIOH1	GPIOH1	SCLK
FPDAT20[1]	GPIOH2	GPIOH2	GPIOH2	GPIOH2	SA0
FPDAT21[1]	GPIOH3	GPIOH3	GPIOH3	GPIOH3	SO
FPDAT22[1]	GPIOH4	GPIOH4	GPIOH4	GPIOH4	SI
FPDAT23[1]	GPIOH5	GPIOH5	GPIOH5	GPIOH5	0 (pull-down)

1. FPDAT[23:18] are not available on the QFP package of the S1D13513.

5.5 LCD Interface Pin Mapping

Table 5-15: LCD Interface Pin Mapping for TFT Panels

S1D13513 Pin Name	Color Active (TFT) Panels														
	Generic TFT			ND-TFD			a-Si TFT			TFT with uWire			Sharp HR-TFT		
	24-bit	18-bit	16-bit	24-bit	18-bit	16-bit	24-bit	18-bit	16-bit	24-bit	18-bit	16-bit	24-bit	18-bit	16-bit
FPFRAME	VSYNC			VSYNC			VSYNC			VSYNC			SPS		
FPLINE	HSYNC			HSYNC			HSYNC			HSYNC			LP		
FPSHIFT	DCK			DCK			DCK			DCK			DCLK		
FPDRDY	ENAB			ENAB			ENAB			ENAB			driven 0		
FPDAT0	R7	R5	R4	R7	R5	R4	R7	R5	R4	R7	R5	R4	R7	R5	R4
FPDAT1	R6	R4	R3	R6	R4	R3	R6	R4	R3	R6	R4	R3	R6	R4	R3
FPDAT2	R5	R3	R2	R5	R3	R2	R5	R3	R2	R5	R3	R2	R5	R3	R2
FPDAT3	G7	G5	G5	G7	G5	G5	G7	G5	G5	G7	G5	G5	G7	G5	G5
FPDAT4	G6	G4	G4	G6	G4	G4	G6	G4	G4	G6	G4	G4	G6	G4	G4
FPDAT5	G5	G3	G3	G5	G3	G3	G5	G3	G3	G5	G3	G3	G5	G3	G3
FPDAT6	B7	B5	B4	B7	B5	B4	B7	B5	B4	B7	B5	B4	B7	B5	B4
FPDAT7	B6	B4	B3	B6	B4	B3	B6	B4	B3	B6	B4	B3	B6	B4	B3
FPDAT8	B5	B3	B2	B5	B3	B2	B5	B3	B2	B5	B3	B2	B5	B3	B2
FPDAT9	R4	R2	R1	R4	R2	R1	R4	R2	R1	R4	R2	R1	R4	R2	R1
FPDAT10	R3	R1	R0	R3	R1	R0	R3	R1	R0	R3	R1	R0	R3	R1	R0
FPDAT11	R2	R0	0	R2	R0	0	R2	R0	0	R2	R0	0	R2	R0	0
FPDAT12	G4	G2	G2	G4	G2	G2	G4	G2	G2	G4	G2	G2	G4	G2	G2
FPDAT13	G3	G1	G1	G3	G1	G1	G3	G1	G1	G3	G1	G1	G3	G1	G1
FPDAT14	G2	G0	G0	G2	G0	G0	G2	G0	G0	G2	G0	G0	G2	G0	G0
FPDAT15	B4	B2	B1	B4	B2	B1	B4	B2	B1	B4	B2	B1	B4	B2	B1
FPDAT16	B3	B1	B0	B3	B1	B0	B3	B1	B0	B3	B1	B0	B3	B1	B0
FPDAT17	B2	B0	0	B2	B0	0	B2	B0	0	B2	B0	0	B2	B0	0
FPDAT18[1]	R1	0	0	R1	0	0	R1	0	0	R1	0	0	R1	0	0
FPDAT19[1]	R0	0	0	R0	0	0	R0	0	0	R0	0	0	R0	0	0
FPDAT20[1]	G1	0	0	G1	0	0	G1	0	0	G1	0	0	G1	0	0
FPDAT21[1]	G0	0	0	G0	0	0	G0	0	0	G0	0	0	G0	0	0
FPDAT22[1]	B1	0	0	B1	0	0	B1	0	0	B1	0	0	B1	0	0
FPDAT23[1]	B0	0	0	B0	0	0	B0	0	0	B0	0	0	B0	0	0
GPIOG0	GPIOG0			XCS			SSTB			LCDCS			PS		
GPIOG1	GPIOG1			SCK			SCLK			SCLK			CLS		
GPIOG2	GPIOG2			A0			driven 0			driven 0			REV		
GPIOG3	GPIOG3			SO			SDATA			SDO			SPL		
GPIOG4	GPIOG4			driven 0			driven 0			driven 0			SPR		

Note

1. The FPDAT[23:18] pins are not available on the QFP package.

Table 5-16: LCD Interface Pin Mapping for Passive Panels

S1D13513 Pin Name	Monochrome Panel	Color Passive Panels
	Single 8-bit	Single 8-bit Format 2
FPFRAME	FPFRAME	FPFRAME
FPLINE	FPLINE	FPLINE
FPSHIFT	FPSHIFT	FPSHIFT
FPDRDY	MOD	MOD
FPDAT0	D0	D0
FPDAT1	D1	D1
FPDAT2	D2	D2
FPDAT3	D3	D3
FPDAT4	D4	D4
FPDAT5	D5	D5
FPDAT6	D6	D6
FPDAT7	D7	D7
FPDAT8	driven 0	driven 0
FPDAT9	driven 0	driven 0
FPDAT10	driven 0	driven 0
FPDAT11	driven 0	driven 0
FPDAT12	driven 0	driven 0
FPDAT13	driven 0	driven 0
FPDAT14	driven 0	driven 0
FPDAT15	driven 0	driven 0
FPDAT16	driven 0	driven 0
FPDAT17	driven 0	driven 0
FPDAT18[1]	GPIOH0	GPIOH0
FPDAT19[1]	GPIOH1	GPIOH1
FPDAT20[1]	GPIOH2	GPIOH2
FPDAT21[1]	GPIOH3	GPIOH3
FPDAT22[1]	GPIOH4	GPIOH4
FPDAT23[1]	GPIOH5	GPIOH5
GPIOG0	GPIOG0	GPIOG0
GPIOG1	GPIOG1	GPIOG1
GPIOG2	GPIOG2	GPIOG2
GPIOG3	GPIOG3	GPIOG3
GPIOG4	GPIOG4	GPIOG4

Note

1. The FPDAT[23:18] pins are not available on the QFP package.

5.6 GPIO Pin Mapping

The GPIO pins are used for various interfaces supported by the S1D13513. The following table summarizes the possible uses for each GPIO pin. Selecting certain combinations of interfaces may not be possible due to the fact that they share the same GPIO pin. For example, the PWM interface is not available at the same time as the Camera2 interface.

Table 5-17 : GPIO Pin Mapping Summary

S1D13513 Pin Name	Controlled By	GPIO Input	Function #1	GPIO Output	Function #2
GPIOA7	REG[0C02h] bits 15-14	00b: GPIOA7	01b: PWMR	10b: GPIOA7	11b: CM2DAT7
GPIOA6	REG[0C02h] bits 13-12	00b: GPIOA6	01b: PWMG	10b: GPIOA6	11b: CM2DAT6
GPIOA5	REG[0C02h] bits 11-10	00b: GPIOA5	01b: PWMB	10b: GPIOA5	11b: CM2DAT5
GPIOA4	REG[0C02h] bits 9-8	00b: GPIOA4	01b: KEYX4	10b: GPIOA4	11b: CM2DAT4
GPIOA3[1]	REG[0C02h] bits 7-6	00b: GPIOA3	01b: KEYX3	10b: GPIOA3	11b: CM2DAT3
GPIOA2[1]	REG[0C02h] bits 5-4	00b: GPIOA2	01b: KEYX2	10b: GPIOA2	11b: CM2DAT2
GPIOA1[1]	REG[0C02h] bits 3-2	00b: GPIOA1	01b: KEYX1	10b: GPIOA1	11b: CM2DAT1
GPIOA0[1]	REG[0C02h] bits 1-0	00b: GPIOA0	01b: KEYX0	10b: GPIOA0	11b: CM2DAT0
GPIOB7	REG[0C06h] bits 15-14	00b: GPIOB7	01b: PWMW	10b: GPIOB7	11b: CMSTROUT
GPIOB6	REG[0C06h] bits 13-12	00b: GPIOB6	01b: SDA	10b: GPIOB6	11b: Reserved
GPIOB5	REG[0C06h] bits 11-10	00b: GPIOB5	01b: SCL	10b: GPIOB5	11b: Reserved
GPIOB4	REG[0C06h] bits 9-8	00b: GPIOB4	01b: KEYY4	10b: GPIOB4	11b: Reserved
GPIOB3[1]	REG[0C06h] bits 7-6	00b: GPIOB3	01b: KEYY3	10b: GPIOB3	11b: CM2VREF
GPIOB2[1]	REG[0C06h] bits 5-4	00b: GPIOB2	01b: KEYY2	10b: GPIOB2	11b: CM2HREF
GPIOB1[1]	REG[0C06h] bits 3-2	00b: GPIOB1	01b: KEYY1	10b: GPIOB1	11b: CM2CLKOUT
GPIOB0[1]	REG[0C06h] bits 1-0	00b: GPIOB0	01b: KEYY0	10b: GPIOB0	11b: CM2CLKIN
GPIOC[7:0]	REG[0C0Ah] bits 15-0	00: GPIOC[7:0]	01b: YUVDATA[7:0]	10: GPIOC[7:0	11b: CM1DAT[7:0]
GPIOD3	REG[0C0Eh] bits 7-6	00: GPIOD3	01b: AUDIN	10: GPIOD3	11b: CM1CLKIN
GPIOD2	REG[0C0Eh] bits 5-4	00: GPIOD2	01b: YUVCLKO	10: GPIOD2	11b: CM1CLKOUT
GPIOD1	REG[0C0Eh] bits 3-2	00: GPIOD1	01b: YUVHSYNC	10: GPIOD1	11b: CM1HREF
GPIOD0	REG[0C0Eh] bits 1-0	00: GPIOD0	01b: YUVVSYNC	10: GPIOD0	11b: CM1VREF
GPIOG[4:0][2]	REG[0C1Ah] bits 9-0	00: GPIOG[4:0]	01b: HR-TFT Support	10: GPIOG[4:0]	11b: Serial Panel Interface Support
GPIOH[5:0][3]	REG[0C1Eh] bits 11-10	00: GPIOH[5:0]	01b: 24-bit TFT Support	10: GPIOH[5:0]	11b: Reserved

🟥	PWM Interface	🟪	YUV Digital Output Interface
🟩	Keypad Interface	🟣	Camera1 Interface
🟦	I2C Interface	🟨	Camera2 Interface

Note

1. The GPIOA[3:0] and GPIOB[3:0] pins are not available on the QFP package. Therefore, any associated functions are also un-available on the QFP package.
2. The GPIOG[4:0] pins are primarily used for extended TFT support, but may be available for use as general purpose IO pins when generic TFT are used. For LCD pin mapping, refer to Section 5.5, “LCD Interface Pin Mapping” on page 48.
3. GPIOH[5:0] are multiplexed on the FPDAT[23:18] pins and may be available for use as general purpose IO pins when a panel that does not require all 24 panel data lines is used. For LCD pin mapping, refer to Section 5.5, “LCD Interface Pin Mapping” on page 48.

5.7 YUV Digital Output Interface Pin Mapping

The YUV Digital Output Interface allows the S1D13513 to output YUV 4:2:2 data to an external video encoder for display on a TV. Note that the YUV Digital Output interface shares some of it's GPIO pins with other interfaces. To determine if specific combinations of interfaces are possible, refer to Figure 5.6 "GPIO Pin Mapping" on page 50.

Table 5-18 : YUV Output Interface Pin Mapping

S1D13513 Pin Name	YUV Digital Output Signal Name	Description
GPIOC[7:0]	YUVDATA[7:0]	YUV Digital data output for external video encoder
GPIOD2	YUVCLKO	Clock output for external video encoder
GPIOD1	YUVHSYNC	Horizontal sync for external video encoder
GPIOD0	YUVVSYNC	Vertical sync for external video encoder

5.8 Camera Interface Pin Mapping

The Camera Interface allows the S1D13513 to receive YUV data (or JPEG data from a JPEG capable camera) from an external camera module. Note that the Camera interface shares some of it's GPIO pins with other interfaces. To determine if specific combinations of interfaces are possible, refer to Figure 5.6 "GPIO Pin Mapping" on page 50.

Table 5-19 : Camera Interface Pin Mapping

S1D13513 Pin Name	Camera Signal Name	Description
GPIOA[7:0]	CM2DAT[7:0]	Data input for Camera2 interface
GPIOB7	CMSTROUT	Strobe signal for the Camera interface
GPIOB3	CM2VREF	Vertical sync input for Camera2 interface
GPIOB2	CM2HREF	Horizontal sync input for Camera2 interface
GPIOB1	CM2CLKOUT	Master clock output for the Camera2 interface
GPIOB0	CM2CLKIN	Pixel clock input for the Camera2 interface
GPIOC[7:0]	CM1DAT[7:0]	Data input for Camera1 interface
GPIOD3	CM1CLKIN	Pixel clock input for the Camera1 interface
GPIOD2	CM1CLKOUT	Master clock output for the Camera1 interface
GPIOD1	CM1HREF	Horizontal sync input for Camera1 interface
GPIOD0	CM1VREF	Vertical sync input for Camera1 interface

Note

The camera2 interface is not available on the QFP package.

Table 5-20 : Camera Interface Pin Mapping (16-bit data bus mode - PBGA Package)

S1D13513 Pin Name	Camera Signal Name	Description
GPIOA[7:0]	CM2DAT[7:0]	Data input for lower 8bit of camera interface
GPIOB7	CMSTROUT	Strobe signal for the Camera interface
GPIOC[7:0]	CM1DAT[7:0]	Data input for higher 8bit of camera interface
GPIOD3	CM1CLKIN	Pixel clock input for the Camera interface
GPIOD2	CM1CLKOUT	Master clock output for the Camera interface
GPIOD1	CM1HREF	Horizontal sync input for Camera interface
GPIOD0	CM1VREF	Vertical sync input for Camera interface

5.9 I2C Interface Pin Mapping

The I2C interface shares some of it's GPIO pins with other interfaces. To determine if specific combinations of interfaces are possible, refer to Figure 5.6 "GPIO Pin Mapping" on page 50.

Table 5-21 : I2C Interface Pin Mapping

S1D13513 Pin Name	I2C Signal Name	Description
GPIOB6	SDA	I2C data
GPIOB5	SCL	I2C data clock

5.10 PWM Interface Pin Mapping

The PWM interface shares some of it's GPIO pins with other interfaces. To determine if specific combinations of interfaces are possible, refer to Figure 5.6 "GPIO Pin Mapping" on page 50.

Table 5-22 : PWM Interface Pin Mapping

S1D13513 Pin Name	PWM Signal Name	Description
GPIOA7	PWMR	PWM Red output
GPIOA6	PWMG	PWM Green output
GPIOA5	PWMB	PWM Blue output
GPIOB7	PWMW	PWM White output
GPIOD3	AUDIN	Digital audio input for PWM circuit

5.11 Keypad Interface Pin Mapping

The keypad interface pins are used to receive input from up to a 5x5 keypad device. The keypad interface shares some of it's GPIO pins with other interfaces. To determine if specific combinations of interfaces are possible, refer to Figure 5.6 "GPIO Pin Mapping" on page 50

Table 5-23 : Keypad Interface Pin Mapping

S1D13513 Pin Name	PWM Signal Name	Description
GPIOA4	KEYX4	X axis pin 4
GPIOA3	KEYX3	X axis pin 3
GPIOA2	KEYX2	X axis pin 2
GPIOA1	KEYX1	X axis pin 1
GPIOA0	KEYX0	X axis pin 0
GPIOB4	KEYY4	Y axis pin 4
GPIOB3	KEYY3	Y axis pin 3
GPIOB2	KEYY2	Y axis pin 2
GPIOB1	KEYY1	Y axis pin 1
GPIOB0	KEYY0	Y axis pin 0

Note

The keypad interface is not available on the QFP package.

6 D.C. Characteristics

Note

1. When applying supply voltages to the S1D13513, Core V_{DD} must be applied to the chip before, or simultaneously with H V_{DD}, or damage to the chip may result.
2. Core V_{DD}, OSC V_{DD}, and PLL V_{DD} must be equal to or lower than H V_{DD}.

Table 6-1 : Absolute Maximum Ratings

Symbol	Parameter	Rating	Units
Core V_{DD}	Supply Voltage	V_{SS} - 0.3 to 2.5	V
H V_{DD}	Supply Voltage	V_{SS} - 0.3 to 4.0	V
OSC V_{DD}	Supply Voltage	V_{SS} - 0.3 to 2.1	V
PLL V_{DD}	Supply Voltage	V_{SS} - 0.3 to 2.1	V
V_{IN}	Input Voltage	V_{SS} - 0.3 to H V_{DD} + 0.5	V
V_{OUT}	Output Voltage	V_{SS} - 0.3 to H V_{DD} + 0.5	V
T_{STG}	Storage Temperature	-65 to 150	° C

Table 6-2 : Recommended Operating Conditions

Symbol	Parameter	Condition	Min	Typ	Max	Units
Core V_{DD}	Supply Voltage	V_{SS} = 0 V	1.65	1.8	1.95	V
H V_{DD1}	Supply Voltage	V_{SS} = 0 V	3.0	3.3	3.6	V
H V_{DD2}	Supply Voltage	V_{SS} = 0 V	3.0	3.3	3.6	V
H V_{DD3}	Supply Voltage	V_{SS} = 0 V	3.0	3.3	3.6	V
H V_{DD4}	Supply Voltage	V_{SS} = 0 V	3.0	3.3	3.6	V
H V_{DD5}	Supply Voltage	V_{SS} = 0 V	3.0	3.3	3.6	V
OSC V_{DD1}	Supply Voltage	V_{SS} = 0 V	1.65	1.8	1.95	V
OSC V_{DD2}	Supply Voltage	V_{SS} = 0 V	1.65	1.8	1.95	V
PLL V_{DD1}	Supply Voltage	V_{SS} = 0 V	1.65	1.8	1.95	V
PLL V_{DD2}	Supply Voltage	V_{SS} = 0 V	1.65	1.8	1.95	V
V_{IN}	Input Voltage		V_{SS}	—	Core V_{DD}	V
			V_{SS}	—	IO V_{DD}	V
T_{OPR}	Operating Temperature		-40	25	85	° C

Table 6-3: Electrical Characteristics for VDD = 3.3V typical

Symbol	Parameter	Condition	Min	Typ	Max	Units
I_{DDS}	Quiescent Current	Quiescent Conditions		150		µA
I_{IZ}	Input Leakage Current	V_I = 0V or V_{DD}	-5	—	5	µA
I_{OZ}	Output Leakage Current	V_O = 0V or V_{DD}	-5	—	5	µA
I_{OH2}	High Level Output Current	V_{OH} = H V_{DD} - 0.4V	-4	—	—	mA
I_{OH3}	High Level Output Current	H V_{DD} = min	-8	—	—	mA
I_{OL2}	Low Level Output Current	V_{OL} = 0.4V	4	—	—	mA
I_{OL3}	Low Level Output Current	H V_{DD} = min	8	—	—	mA
V_{IH}	High Level Input Voltage	LVCMOS level, H V_{DD} = max	2.2	—	H V_{DD} + 0.3	V
V_{IL}	Low Level Input Voltage	LVCMOS level, H V_{DD} = min	-0.3	—	0.8	V
VT_+	Positive Trigger Voltage	LVCMOS Schmitt	1.4	—	2.7	V
VT_-	Negative Trigger Voltage	LVCMOS Schmitt	0.6	—	1.8	V
V_H	Hysteresis Voltage	LVCMOS Schmitt	0.3	—	—	V
R_{PU}	Pull-up Resistance	V_I = 0V, Type 1	25	50	120	kΩ
		V_I = 0V, Type 2	50	100	240	kΩ
R_{PD}	Pull-down Resistance	V_I = H V_{DD}, Type 1	25	50	120	kΩ
		V_I = H V_{DD}, Type 2	50	100	240	kΩ
C_I	Input Pin Capacitance	F = 1MHz, H V_{DD} = 0V	—	—	8	pF
C_O	Output Pin Capacitance	F = 1MHz, H V_{DD} = 0V	—	—	8	pF
C_{IO}	Bi-Directional Pin Capacitance	F = 1MHz, H V_{DD} = 0V	—	—	8	pF

7 A.C. Characteristics

Conditions: IO V_{DD} = 3.3V ± 10%
T_A = -25° C to 85° C
T_{rise} and T_{fall} for all inputs must be $\leq$ 5 ns (10% ~ 90%)
C_L = 50pF (CPU Interface), unless noted
C_L = 0pF (LCD Panel Interface)
C_L = 15pF (Display Memory Interface)

7.1 Clock Timing

7.1.1 Input Clocks

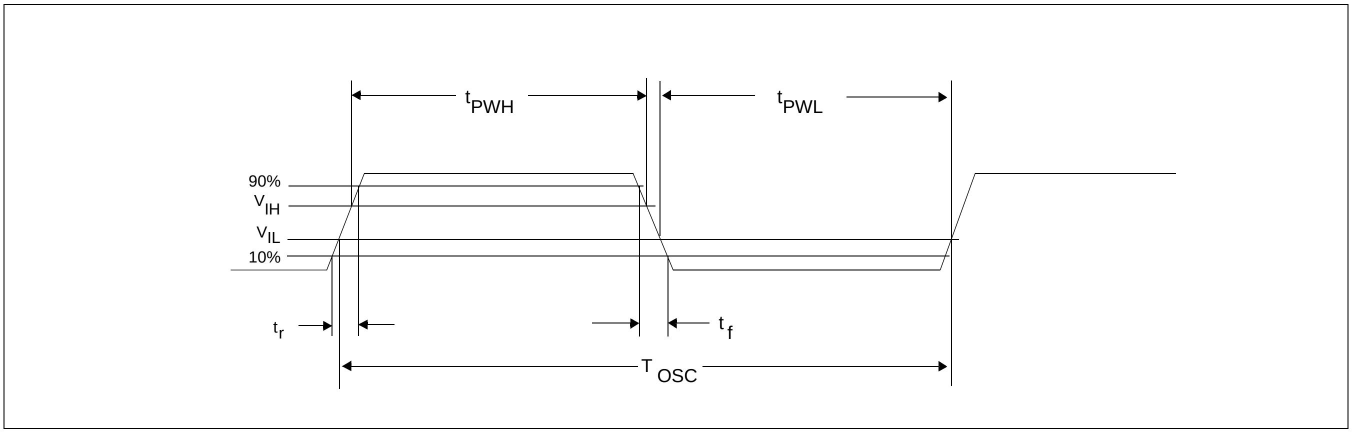


Figure 7-1: Clock Requirements for OSC1/OSC2/CLKI3/BUSCLK

Table 7-1 : Clock Requirements for OSC1/OSC2/CLKI3/BUSCLK when used as Clock Input

Symbol	Parameter	Min	Typ	Max	Units
f_{OSC1}	Input Clock Frequency for OSC1	5	—	65	MHz
T_{OSC1}	Input Clock Period for OSC1	—	$1/f_{OSC1}$	—	ns
f_{OSC2}	Input Clock Frequency for OSC2	5	—	27	MHz
T_{OSC2}	Input Clock Period for OSC2	—	$1/f_{OSC2}$	—	ns
f_{CLKI3}	Input Clock Frequency for CLKI3	5	—	100	MHz
T_{CLKI3}	Input Clock Period for CLKI3	—	$1/f_{CLKI3}$	—	ns
f_{BUSCLK}	Input Clock Frequency for BUSCLK (see Note)	5	—	65	MHz
T_{BUSCLK}	Input Clock Period for BUSCLK (see Note)	—	$1/f_{BUSCLK}$	—	ns
t_{PWH}	Input Clock Pulse Width High	0.4	—	0.6	T_{OSC}
t_{PWL}	Input Clock Pulse Width Low	0.4	—	0.6	T_{OSC}
t_f	Input Clock Fall Time (10% - 90%)	—	—	0.2	T_{OSC}
t_r	Input Clock Rise Time (10% - 90%)	—	—	0.2	T_{OSC}

Note

If a host interface that uses BUSCLK is selected (CNF6=0), BUSCLK must remain on continuously while the S1D13513 is in normal operation mode.

Table 7-2 : Clock Requirements for OSC1/OSC2 when used as Crystal Oscillator Input

Symbol	Parameter	Min	Typ	Max	Units
f_{OSC1}	Input Clock Frequency for OSC1	5	—	20	MHz
T_{OSC1}	Input Clock Period for OSC1	—	$1/f_{OSC1}$	—	ns
f_{OSC2}	Input Clock Frequency for OSC2	5	—	27	MHz
T_{OSC2}	Input Clock Period for OSC2	—	$1/f_{OSC2}$	—	ns
t_{PWH}	Input Clock Pulse Width High	0.4	—	0.6	T_{OSC}
t_{PWL}	Input Clock Pulse Width Low	0.4	—	0.6	T_{OSC}
t_f	Input Clock Fall Time (10% - 90%)	—	—	0.2	T_{OSC}
t_r	Input Clock Rise Time (10% - 90%)	—	—	0.2	T_{OSC}

Note

The maximum allowable clock jitter is 300 ps.

7.1.2 Internal Clocks

The following section provides min/max values for some of the S1D13513 internal clocks. For further information on the internal clocks, refer to Section 9, “Clocks” on page 124.

Table 7-3: Internal Clock Requirements

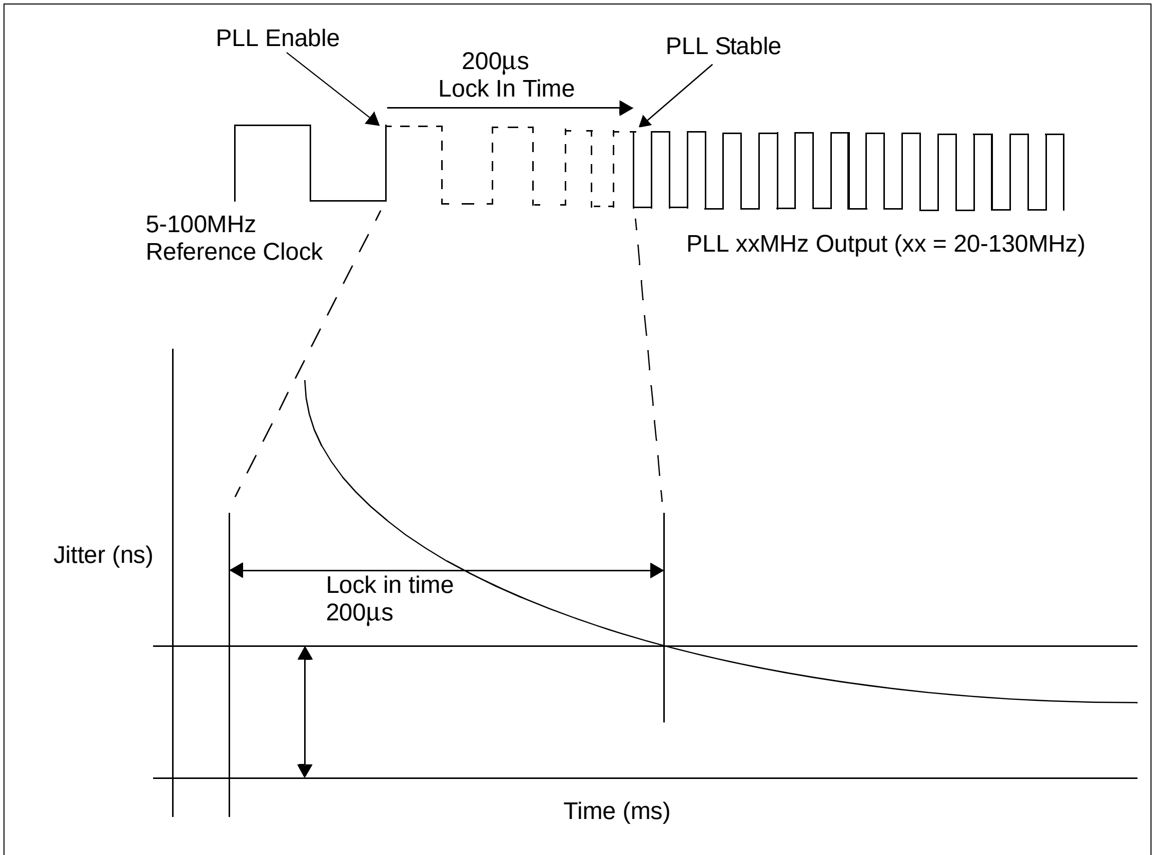
Symbol	Parameter	Min	Max	Units
f_{BCLK}	Bus Clock Frequency	20	50 (Note 1,2)	MHz
$f_{SDRAMCLK}$	SDRAM Clock Frequency	—	100	MHz
f_{SYSCLK}	System Clock Frequency	20	50	MHz

1. The S1D13513 maximum Bus Clock frequency is 50MHz. However, not all Host bus interfaces support a maximum frequency of 50MHz. For timing details for each Host bus interface, refer to Section 7.4, “Host Bus Interface Timing” on page 63.
2. If a host interface that uses BUSCLK is selected (CNF6=0), BUSCLK must remain on continuously while the S1D13513 is in normal operation mode.

7.1.3 PLL Clock

The PLL circuit is an analog circuit and is very sensitive to noise on the input clock waveform or the power supply. Noise on the clock or the supplied power may cause the operation of the PLL circuit to become unstable or increase the jitter.

Due to these noise constraints, it is highly recommended that the power supply traces or the power plane for the PLL be isolated from those of other power supplies. Filtering should also be used to keep the power as clean as possible. The jitter of the input clock waveform should be as small as possible.



The PLL frequency will ramp between the OFF state and the programmed frequency.
To guarantee the lowest possible clock jitter, 200µs is required for stabilization.

Figure 7-2: PLL Start-Up Time

Table 7-4: PLL Clock Requirements

Symbol	Parameter	Min	Max	Units
f_{PLL}	PLL output clock frequency	20	130	MHz
t_{PStal}	PLL output stable time	—	200	µs

7.2 Power Supply Sequence

7.2.1 Power Supply Structure

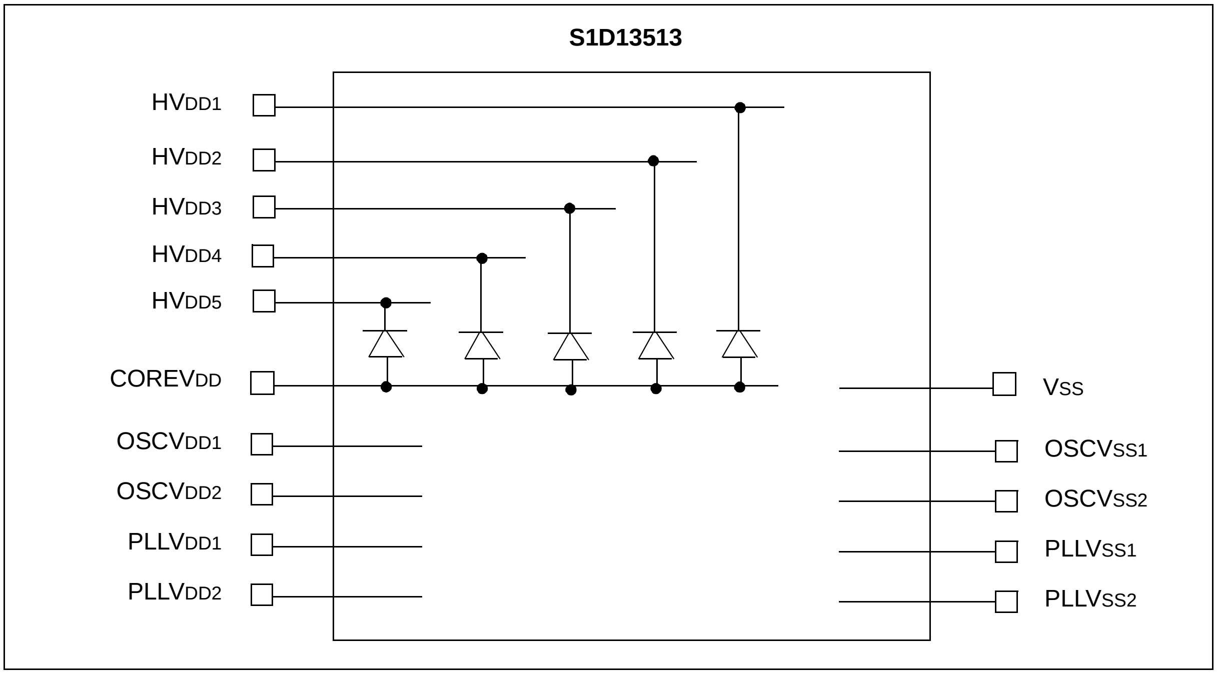


Figure 7-3: Internal Power Structure

7.2.2 Power-On Sequence

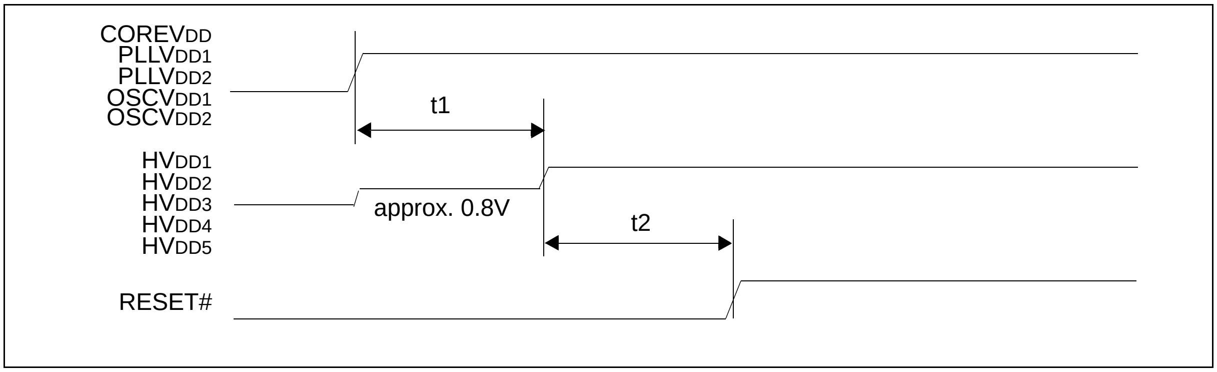


Figure 7-4: Power-On Sequence

Table 7-5: Power-On Sequence

Symbol	Parameter	Min	Max	Units
t1	HV_{DD1} ~ HV_{DD5} on delay from $COREV_{DD}$, $OSCV_{DD1}$, $OSCV_{DD2}$, $PLLV_{DD1}$, $PLLV_{DD2}$ on	0	500	ms
t2	RESET# deasserted from HV_{DD1} ~ HV_{DD5} on	50	—	ns

7.2.3 Power-Off Sequence

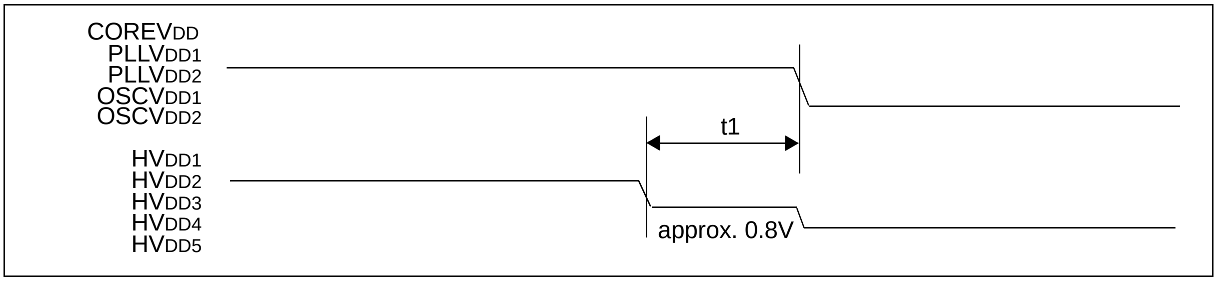


Figure 7-5: Power-Off Sequence

Table 7-6: Power-Off Sequence

Symbol	Parameter	Min	Max	Units
t1	$COREV_{DD}$, $OSCV_{DD1}$, $OSCV_{DD2}$, $PLLV_{DD1}$, $PLLV_{DD2}$ off delay from HV_{DD1} ~ HV_{DD5} off	0	500	ms

7.3 RESET# Timing

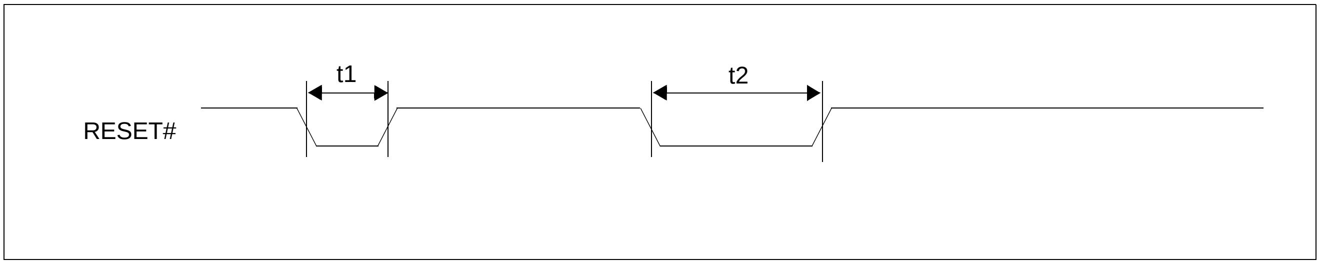


Figure 7-6 S1D13513 RESET# Timing

Table 7-7 S1D13513 RESET# Timing

Symbol	Parameter	Min	Max	Units
t1	Reset Pulse Width to be ignored	—	2	ns
t2	Active Reset Pulse Width	50	—	ns

7.4 Host Bus Interface Timing

7.4.1 Direct/Indirect 80 Type 1

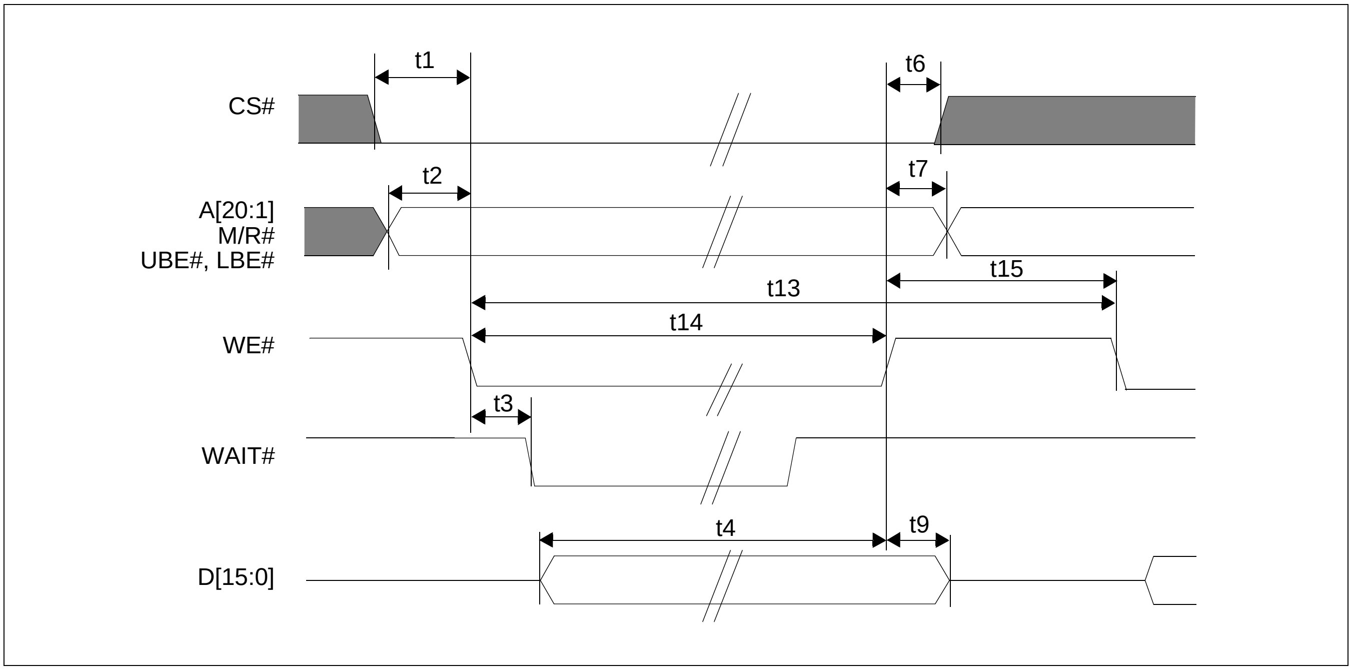


Figure 7-7: Direct/Indirect 80 Type 1 Host Interface Write Timing

Table 7-8 : Direct/Indirect 80 Type 1 Host Interface Write Timing

Symbol	Parameter	Min	Max	Units
t1	CS# setup time to WE# falling edge	5	—	ns
t2	A[20:1], M/R#, UBE#, LBE# setup time to WE# falling edge	5	—	ns
t3	WE# falling edge to WAIT# driven low	—	19	ns
t4	D[15:0] setup time to WE# rising edge	15	—	ns
t6	WE# rising edge to CS# hold time	4	—	ns
t7	WE# rising edge to A[20:1], M/R#, UBE#, LBE# hold time	4	—	ns
t9	D[15:0] hold time from WE# rising edge	5	—	ns
t13	WE# cycle time	4.5	—	Ts (Note 1)
t14	WE# pulse active time	3	—	Ts
t15	WE# pulse inactive time	1.5	—	Ts

1. Ts = System clock period

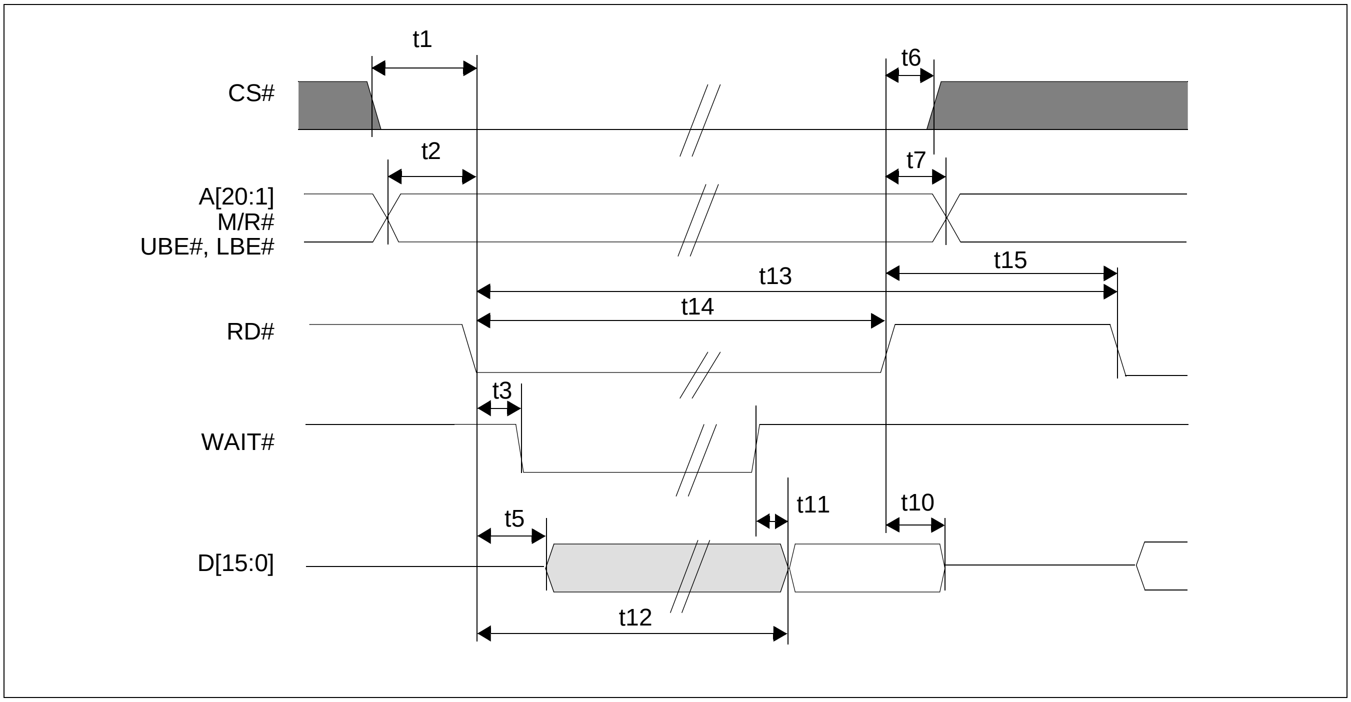


Figure 7-8: Direct/Indirect 80 Type 1 Host Interface Read Timing

Table 7-9 : Direct/Indirect 80 Type 1 Host Interface Read Timing

Symbol	Parameter	Min	Max	Units
t1	CS# setup time to RD# falling edge	5	—	ns
t2	A[20:1], M/R#, UBE#, LBE# setup time to RD# falling edge	5	—	ns
t3	RD# falling edge to WAIT# driven low	—	19	ns
t5	RD# falling edge to D[15:0] driven	4	—	ns
t6	RD# rising edge to CS# hold time	4	—	ns
t7	RD# rising edge to A[20:1], M/R#, UBE#, LBE# hold time	4	—	ns
t10	D[15:0] hold time from RD# rising edge	1	10	ns
t11	WAIT# rising edge to valid DATA if WAIT# asserted	—	10	ns
t12	RD# falling edge to valid Data if WAIT# is NOT asserted	—	20	ns
t13	RD# cycle time	4.5	—	Ts (Note 1)
t14	RD# pulse active time	3	—	Ts
t15	RD# pulse inactive time	1.5	—	Ts

1. Ts = System clock period

7.4.2 Direct/Indirect 80 Type 2

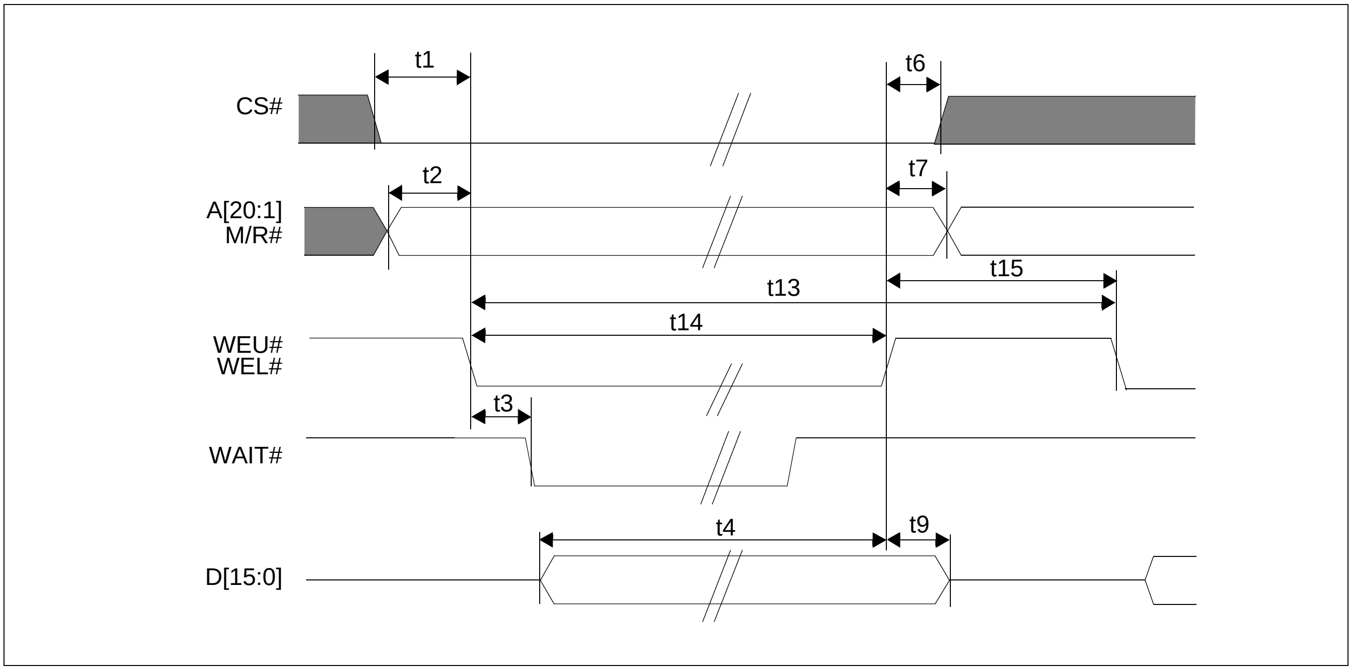


Figure 7-9: Direct/Indirect 80 Type 2 Host Interface Write Timing

Table 7-10 : Direct/Indirect 80 Type 2 Host Interface Write Timing

Symbol	Parameter	Min	Max	Units
t1	CS# setup time to WEU#, WEL# falling edge	9	—	ns
t2	A[20:1], M/R# setup time to WEU#, WEL# falling edge	9	—	ns
t3	WEU#, WEL# falling edge to WAIT# driven low	—	19	ns
t4	D[15:0] setup time to WEU#, WEL# rising edge	15	—	ns
t6	WEU#, WEL# rising edge to CS# hold time	4	—	ns
t7	WEU#, WEL# rising edge to A[20:1], M/R# hold time	4	—	ns
t9	D[15:0] hold time from WEU#, WEL# rising edge	5	—	ns
t13	WEU#, WEL# cycle time	4.5	—	Ts (Note 1)
t14	WEU#, WEL# pulse active time	3	—	Ts
t15	WEU#, WEL# pulse inactive time	1.5	—	Ts

1. Ts = System clock period

Note

Big Endian Mode (CNF5=1) is not supported for Direct 80 Type 2 interfaces.

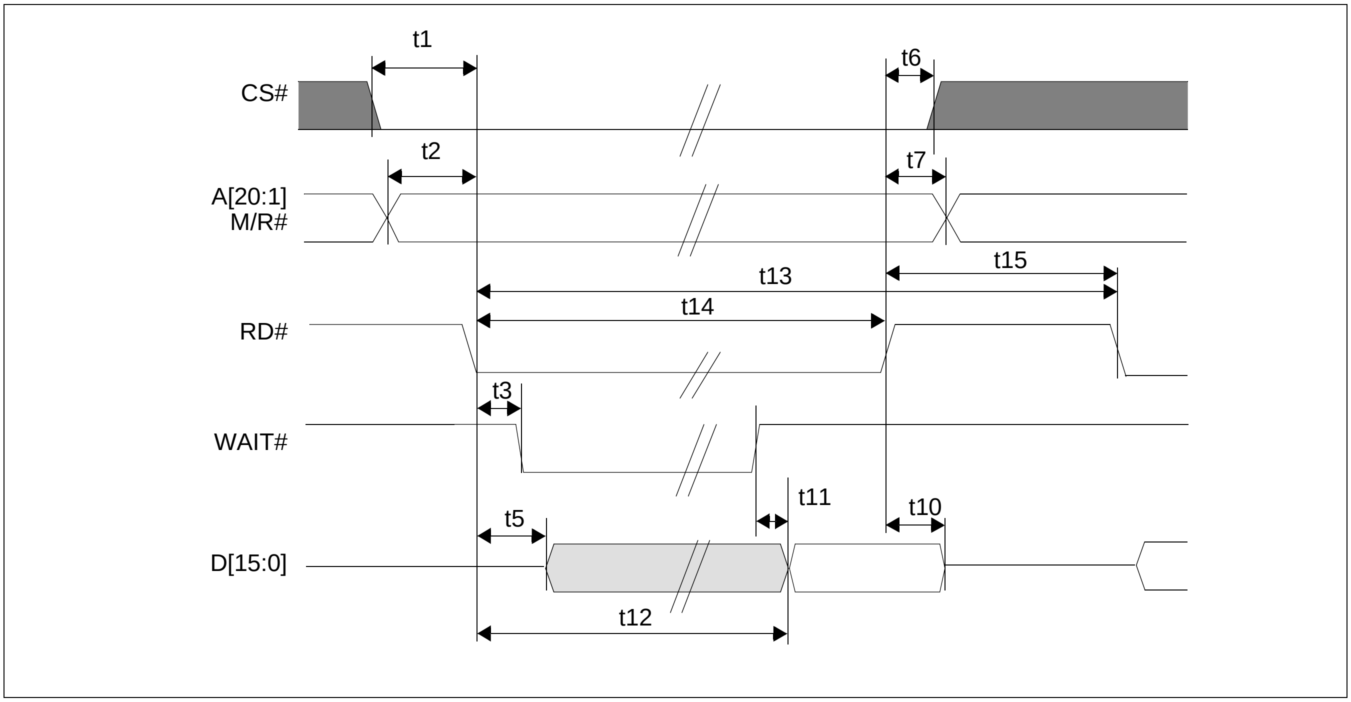


Figure 7-10: Direct/Indirect 80 Type 2 Host Interface Read Timing

Table 7-11 : Direct/Indirect 80 Type 2 Host Interface Read Timing

Symbol	Parameter	Min	Max	Units
t1	CS# setup time to RD# falling edge	9	—	ns
t2	A[20:1], M/R# setup time to RD# falling edge	9	—	ns
t3	RD# falling edge to WAIT# driven low	—	19	ns
t5	RD# falling edge to D[15:0] driven	4	—	ns
t6	RD# rising edge to CS# hold time	4	—	ns
t7	RD# rising edge to A[20:1], M/R# hold time	4	—	ns
t10	D[15:0] hold time from RD# rising edge	1	10	ns
t11	WAIT# rising edge to valid DATA if WAIT# asserted	—	10	ns
t12	RD# falling edge to valid Data if WAIT# is NOT asserted	—	20	ns
t13	RD# cycle time	4.5	—	Ts (Note 1)
t14	RD# pulse active time	3	—	Ts
t15	RD# pulse inactive time	1.5	—	Ts

1. Ts = System clock period

Note

Big Endian Mode (CNF5=1) is not supported for Direct 80 Type 2 interfaces.

7.4.3 Direct/Indirect 68

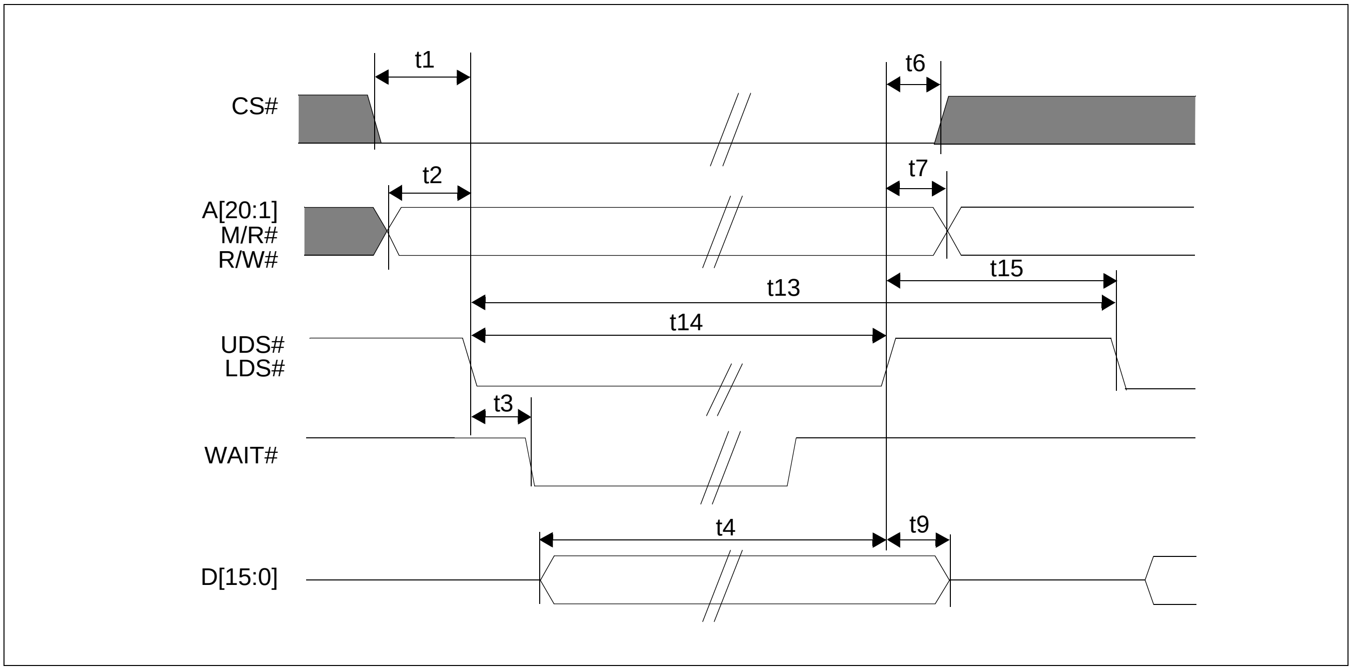


Figure 7-11: Direct/Indirect 68 Host Interface Write Timing

Table 7-12 : Direct/Indirect 68 Host Interface Write Timing

Symbol	Parameter	Min	Max	Units
t1	CS# setup time to UDS#, LDS# falling edge	9	—	ns
t2	A[20:1], M/R#, R/W# setup time to UDS#, LDS# falling edge	9	—	ns
t3	UDS#, LDS# falling edge to WAIT# driven low	—	19	ns
t4	D[15:0] setup time to UDS#, LDS# rising edge	15	—	ns
t6	UDS#, LDS# rising edge to CS# hold time	4	—	ns
t7	UDS#, LDS# rising edge to A[20:1], M/R#, R/W# hold time	4	—	ns
t9	D[15:0] hold time from UDS#, LDS# rising edge	5	—	ns
t13	UDS#, LDS# cycle time	4.5	—	Ts (Note 1)
t14	UDS#, LDS# pulse active time	3	—	Ts
t15	UDS#, LDS# pulse inactive time	1.5	—	Ts

1. Ts = System clock period

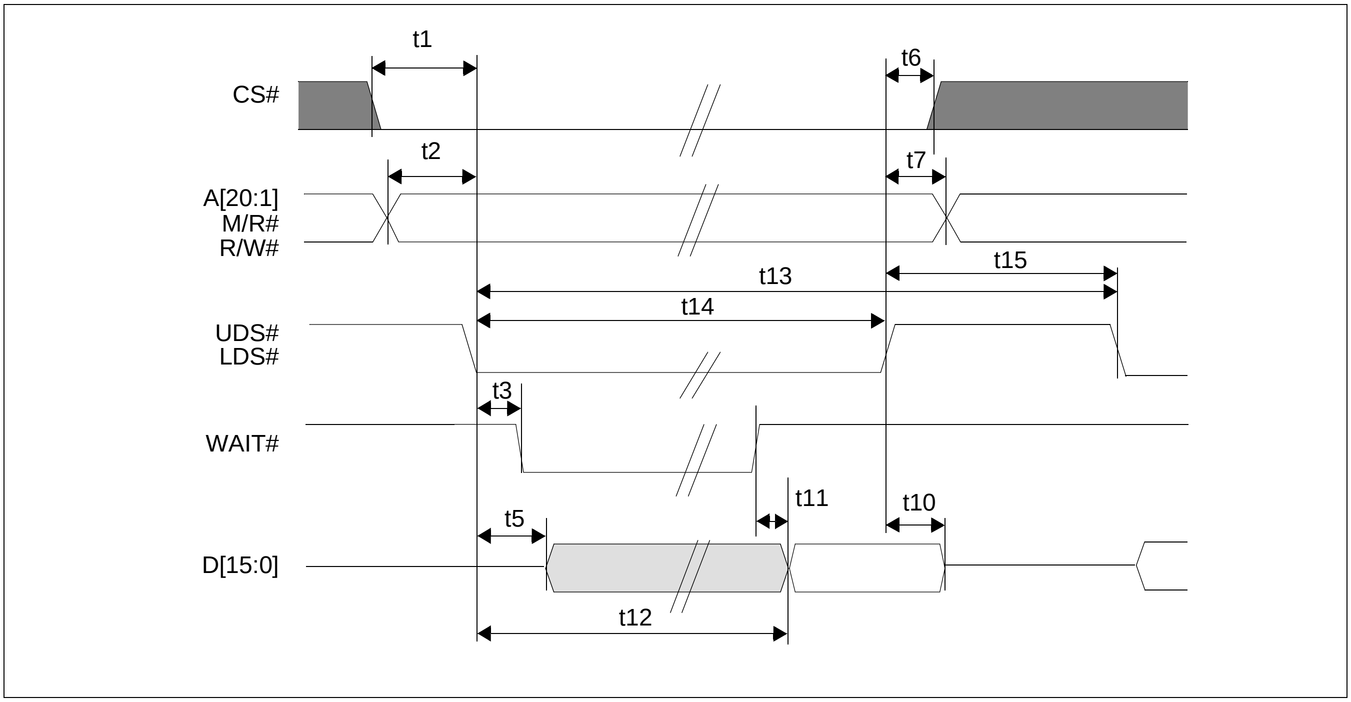


Figure 7-12: Direct/Indirect 68 Host Interface Read Timing

Table 7-13 : Direct/Indirect 68 Host Interface Read Timing

Symbol	Parameter	Min	Max	Units
t1	CS# setup time to UDS#, LDS# falling edge	9	—	ns
t2	A[20:1], M/R#, R/W setup time to UDS#, LDS# falling edge	9	—	ns
t3	UDS#, LDS# falling edge to WAIT# driven low	—	19	ns
t5	UDS#, LDS# falling edge to D[15:0] driven	4	—	ns
t6	UDS#, LDS# rising edge to CS# hold time	4	—	ns
t7	UDS#, LDS# rising edge to A[20:1], M/R#, R/W hold time	4	—	ns
t10	D[15:0] hold time from UDS#, LDS# rising edge	1	10	ns
t11	WAIT# rising edge to valid DATA if WAIT# asserted	—	10	ns
t12	UDS#, LDS# falling edge to valid Data if WAIT# is NOT asserted	—	20	ns
t13	UDS#, LDS# cycle time	4.5	—	Ts (Note 1)
t14	UDS#, LDS# pulse active time	3	—	Ts
t15	UDS#, LDS# pulse inactive time	1.5	—	Ts

1. Ts = System clock period

7.4.4 Generic (i.e. C33)

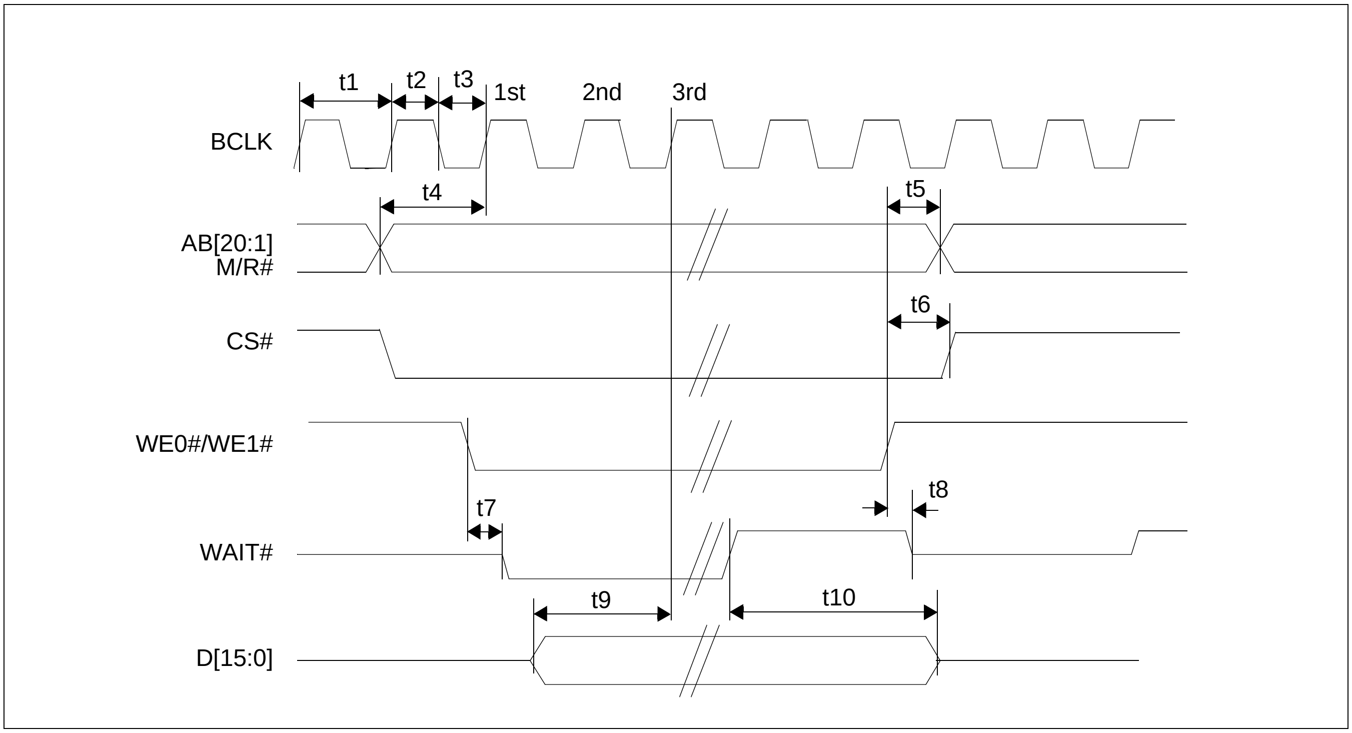


Figure 7-13: Generic Host Interface Write Timing

Table 7-14 :   Generic Host Interface Write Timing

Symbol	Parameter	Min	Max	Units
f_{BCLK}	Clock frequency	—	50	MHz
t1	Clock period	$1/f_{BCLK}$	—	ns
t2	Clock pulse width high	6	—	ns
t3	Clock pulse width low	6	—	ns
t4	A[20:1], M/R# setup to first BCLK where CS# = 0 and WE0#, WE1# = 0	4	—	ns
t5	A[20:1], M/R# hold from rising edge of WE0#, WE1#	0	—	ns
t6	CS# hold from rising edge of WE0#, WE1#	0	—	ns
t7	Falling edge of WE0#, WE1# to WAIT# driven low	3	13	ns
t8	Rising edge of WE0#, WE1# to WAIT# tristate	3	11	ns
t9	D[15:0] setup to third BCLK where CS# = 0 and WE0#, WE1# = 0	0	—	ns
t10	D[15:0] hold	0	—	ns

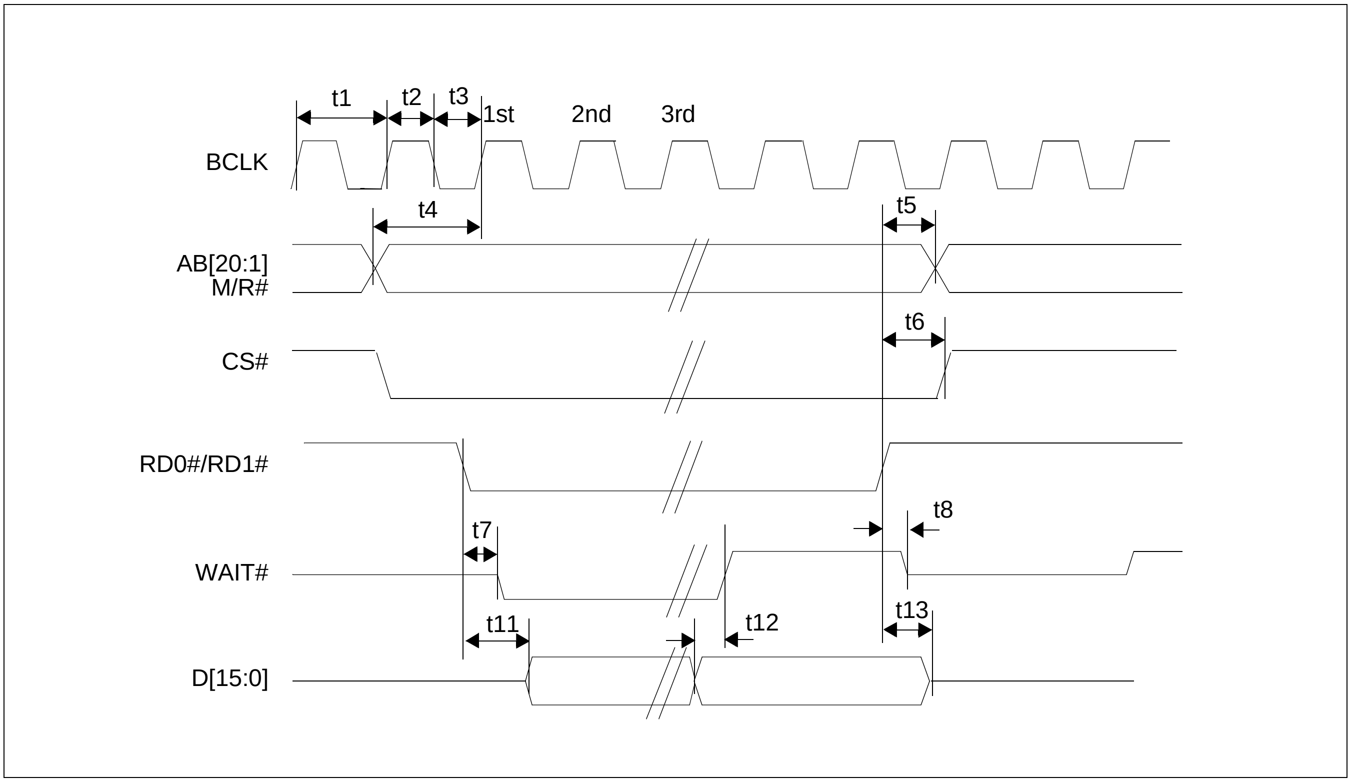


Figure 7-14: Generic Host Interface Read Timing

Table 7-15 : Generic Host Interface Read Timing

Symbol	Parameter	Min	Max	Units
f_{BCLK}	Clock frequency	—	50	MHz
t1	Clock period	$1/f_{BCLK}$	—	ns
t2	Clock pulse width high	6	—	ns
t3	Clock pulse width low	6	—	ns
t4	A[20:1], M/R# setup to first BCLK where CS# = 0 and RD0#, RD1# = 0	4	—	ns
t5	A[20:1], M/R# hold from rising edge of RD0#, RD1#	0	—	ns
t6	CS# hold from rising edge of RD0#, RD1#	0	—	ns
t7	Falling edge of RD0#, RD1# to WAIT# driven low	3	13	ns
t8	Rising edge of RD0#, RD1# to WAIT# tristate	3	11	ns
t11	Falling edge RD0#, RD1# to D[15:0] driven	3	—	ns
t12	D[15:0] setup to rising edge of WAIT#	0	—	ns
t13	Rising edge of RD0#, RD1# to D[15:0] tristate	3	11	ns

7.4.5 Renesas SH4

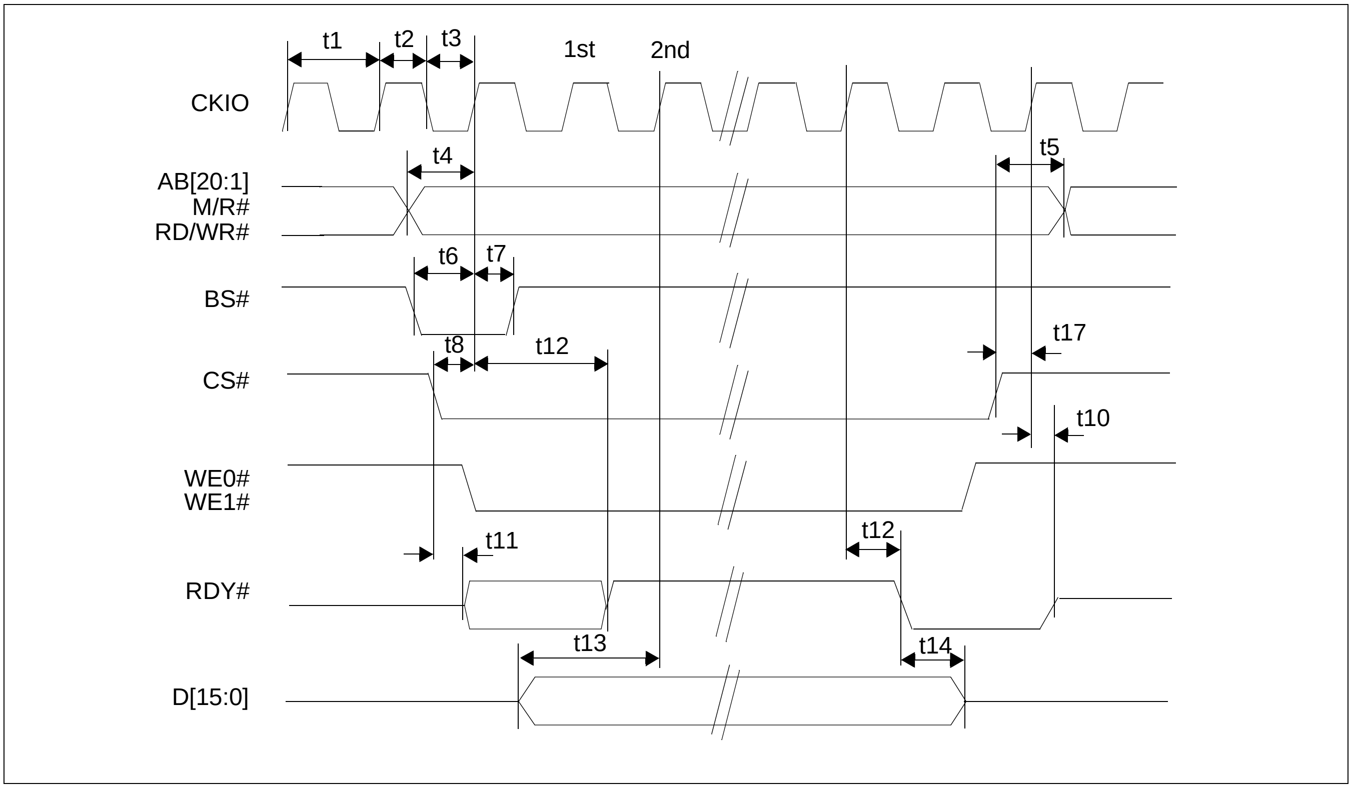


Figure 7-15: Renesas SH4 Host Interface Write Timing

Table 7-16 : Renesas SH4 Host Interface Write Timing

Symbol	Parameter	Min	Max	Units
f_{CKIO}	Clock frequency	—	50	MHz
t1	Clock period	$1/f_{CKIO}$	—	ns
t2	Clock pulse width high	7	—	ns
t3	Clock pulse width low	7	—	ns
t4	A[20:1], M/R#, RD/WR# setup to CKIO	4	—	ns
t5	A[20:1], M/R#, RD/WR# hold from CS#	0	—	ns
t6	BS# setup	4	—	ns
t7	BS# hold	3	—	ns
t8	CS# setup	4	—	ns
t10	CKIO to RDY# tristate	3	12	ns
t11	Falling edge CS# to RDY# driven	3	11	ns
t12	CKIO to RDY# delay	4	13	ns
t13	D[15:0] setup to 2nd CKIO after BS#	0	—	ns
t14	D[15:0] hold	5	—	ns
t17	CS# high setup to CKIO	4	—	ns

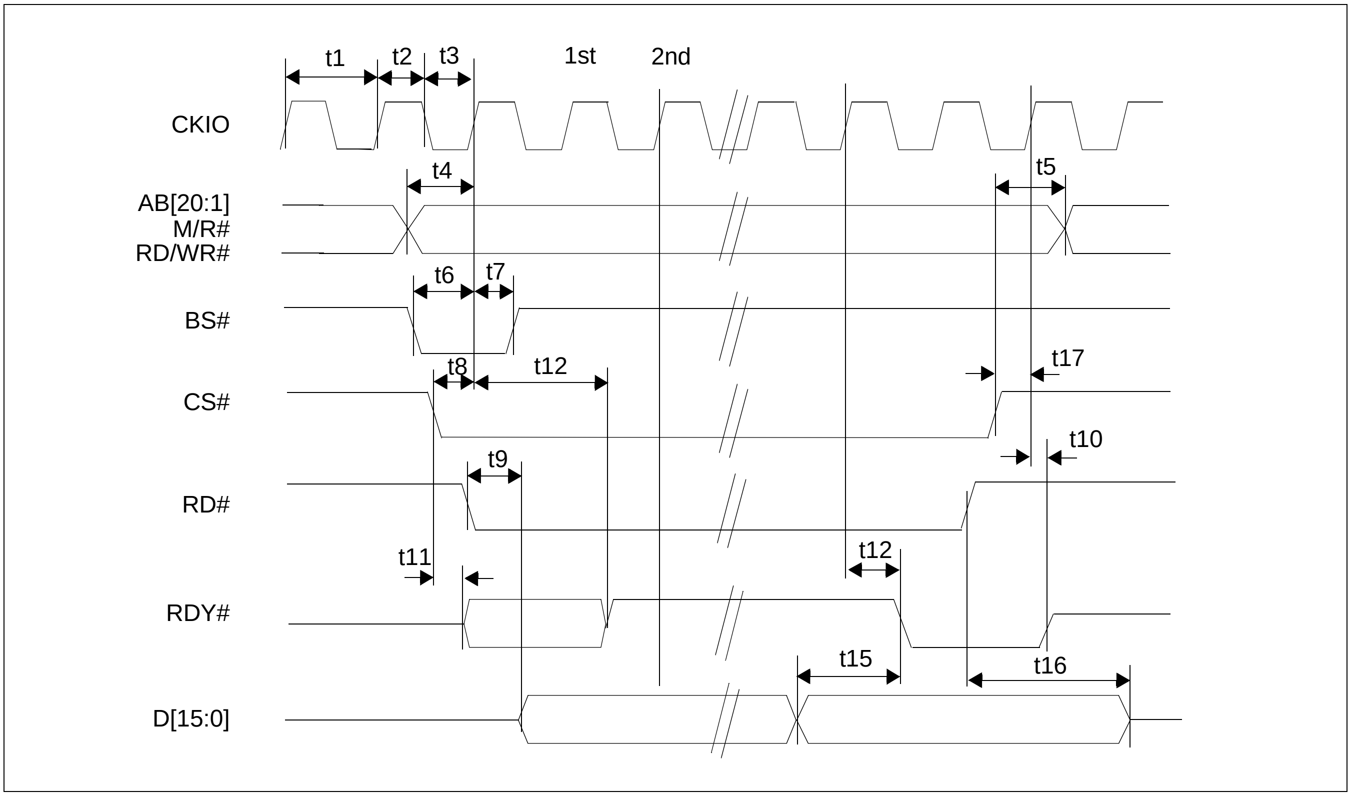


Figure 7-16: Renesas SH4 Host Interface Read Timing

Table 7-17 : Renesas SH4 Host Interface Read Timing

Symbol	Parameter	Min	Max	Units
f_{CKIO}	Clock frequency	—	50	MHz
t1	Clock period	$1/f_{CKIO}$	—	ns
t2	Clock pulse width high	7	—	ns
t3	Clock pulse width low	7	—	ns
t4	A[20:1], M/R#, RD/WR# setup to CKIO	4	—	ns
t5	A[20:1], M/R#, RD/WR# hold from CS#	0	—	ns
t6	BS# setup	4	—	ns
t7	BS# hold	3	—	ns
t8	CS# setup	4	—	ns
t9	Falling edge of RD# to D[15:0] driven	3	—	ns
t10	CKIO to RDY# tristate	3	12	ns
t11	Falling edge of CS# to WAIT# driven	3	11	ns
t12	CKIO to RDY# delay	4	13	ns
t15	D[15:0] valid to RDY# falling edge	0	—	ns
t16	Rising edge of RD# to D[15:0] tristate	3	12	ns
t17	CS# high setup to CKIO	4	—	ns

7.4.6 Renesas SH3

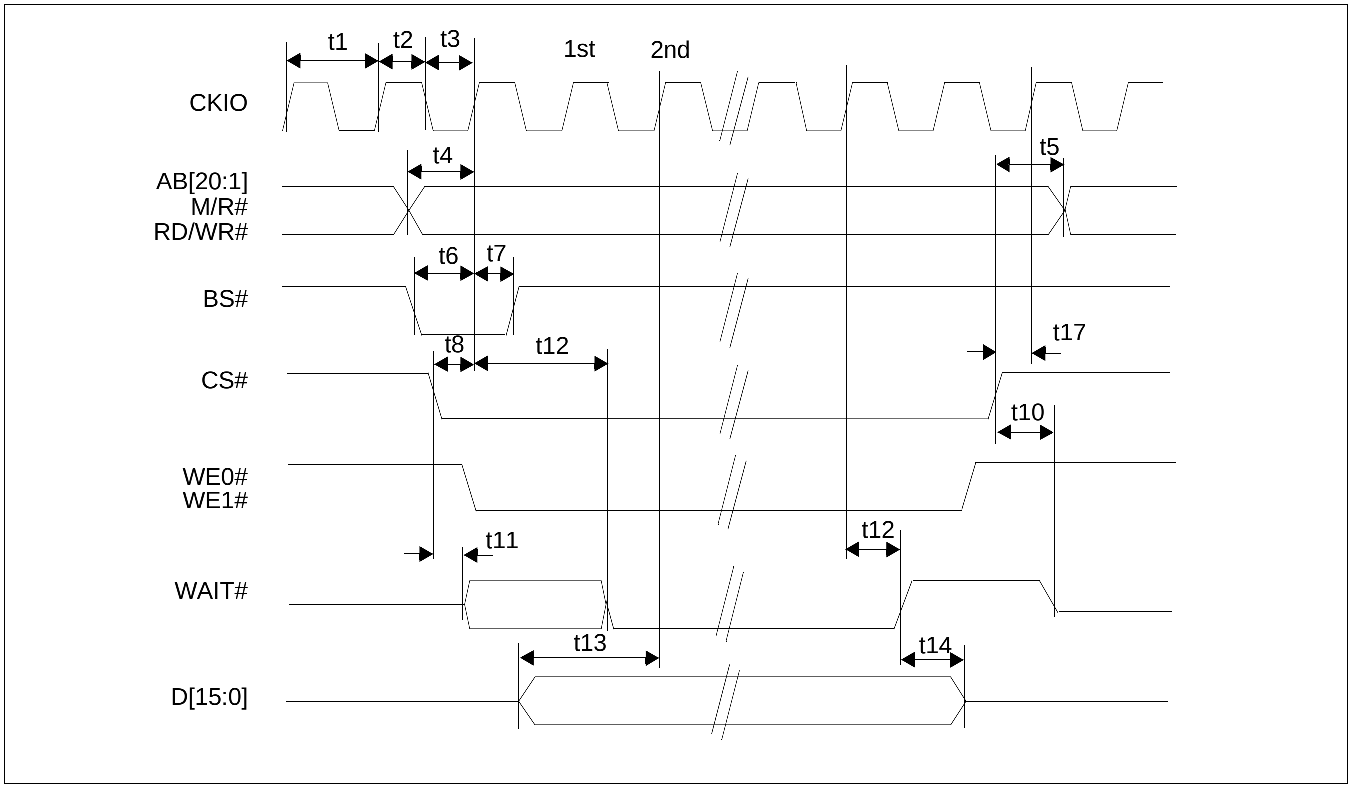


Figure 7-17: Renesas SH3 Host Interface Write Timing

Table 7-18 : Renesas SH3 Host Interface Write Timing

Symbol	Parameter	Min	Max	Units
f_{CKIO}	Clock frequency	—	50	MHz
t1	Clock period	$1/f_{CKIO}$	—	ns
t2	Clock pulse width high	7	—	ns
t3	Clock pulse width low	7	—	ns
t4	A[20:1], M/R#, RD/WR# setup to CKIO	4	—	ns
t5	A[20:1], M/R#, RD/WR# hold from CS#	0	—	ns
t6	BS# setup	4	—	ns
t7	BS# hold	3	—	ns
t8	CS# setup	4	—	ns
t10	Rising edge CS# to WAIT# tristate	3	12	ns
t11	Falling edge CS# to WAIT# driven	3	16	ns
t12	CKIO to WAIT# delay	4	15	ns
t13	D[15:0] setup to 2nd CKIO after BS#	0	—	ns
t14	D[15:0] hold	5	—	ns
t17	CS# high setup to CKIO	4	—	ns

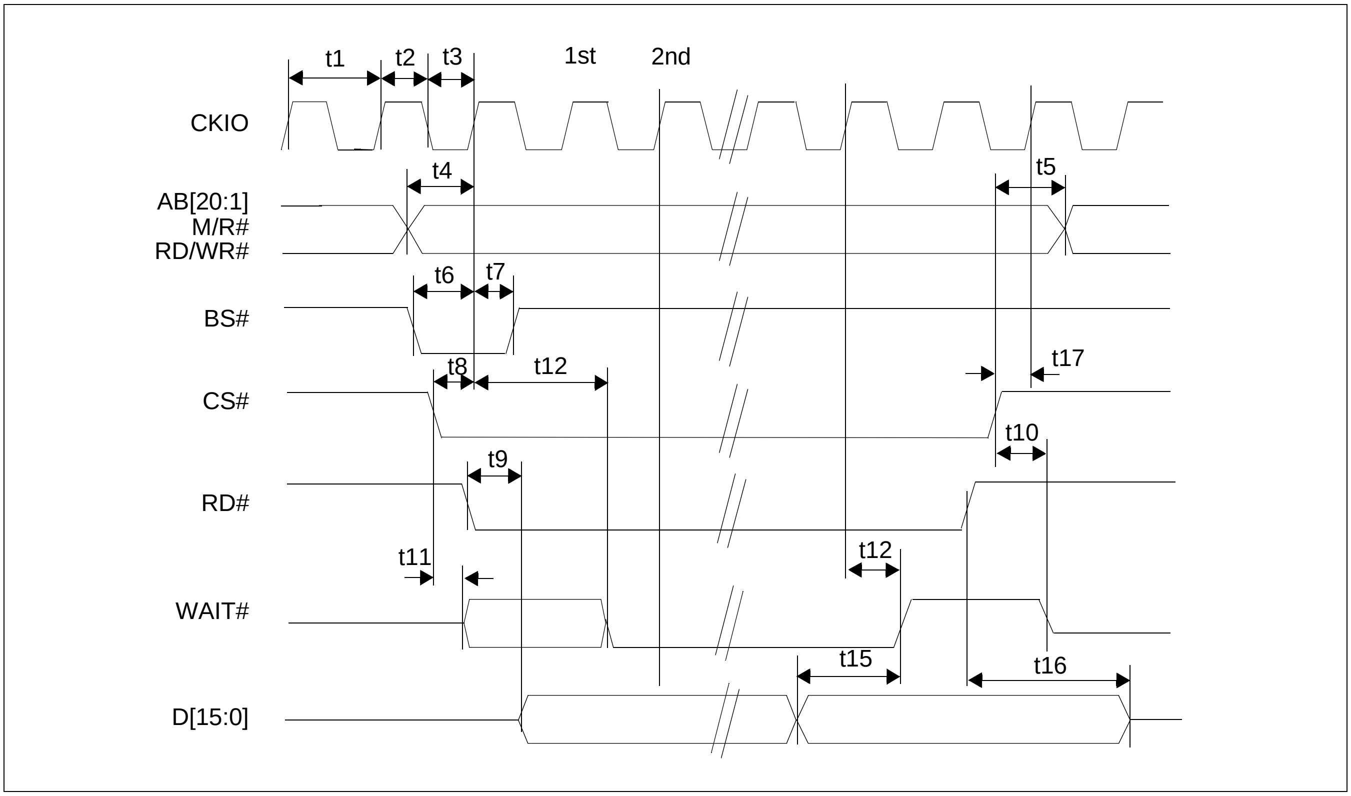


Figure 7-18: Renesas SH3 Host Interface Read Timing

Table 7-19 : Renesas SH3 Host Interface Read Timing

Symbol	Parameter	Min	Max	Units
f_{CKIO}	Clock frequency	—	50	MHz
t1	Clock period	$1/f_{CKIO}$	—	ns
t2	Clock pulse width high	7	—	ns
t3	Clock pulse width low	7	—	ns
t4	A[20:1], M/R#, RD/WR# setup to CKIO	4	—	ns
t5	A[20:1], M/R#, RD/WR# hold from CS#	0	—	ns
t6	BS# setup	4	—	ns
t7	BS# hold	3	—	ns
t8	CS# setup	4	—	ns
t9	Falling edge RD# to D[15:0] driven	3	—	ns
t10	Rising edge CS# to WAIT# tristate	3	12	ns
t11	Falling edge CS# to WAIT# driven	3	16	ns
t12	CKIO to WAIT# delay	4	15	ns
t15	D[15:0] valid to WAIT# rising edge	0	—	ns
t16	Rising edge RD# to D[15:0] tristate	3	12	ns
t17	CS# high setup to CKIO	4	—	ns

Note

For SH3 Active Low WAIT# always driven mode (CNF[4:0] = 11001 or 11101), the output of WAIT# is low during the initialization process. WAIT# will become valid after the system clock is on. This limitation is only valid for revisions 00h and 01h of the S1D13513. The product revision can be checked by reading REG[0000h] bits 15-8.

7.4.7 MIPS/ISA (i.e. NEC VR41xx)

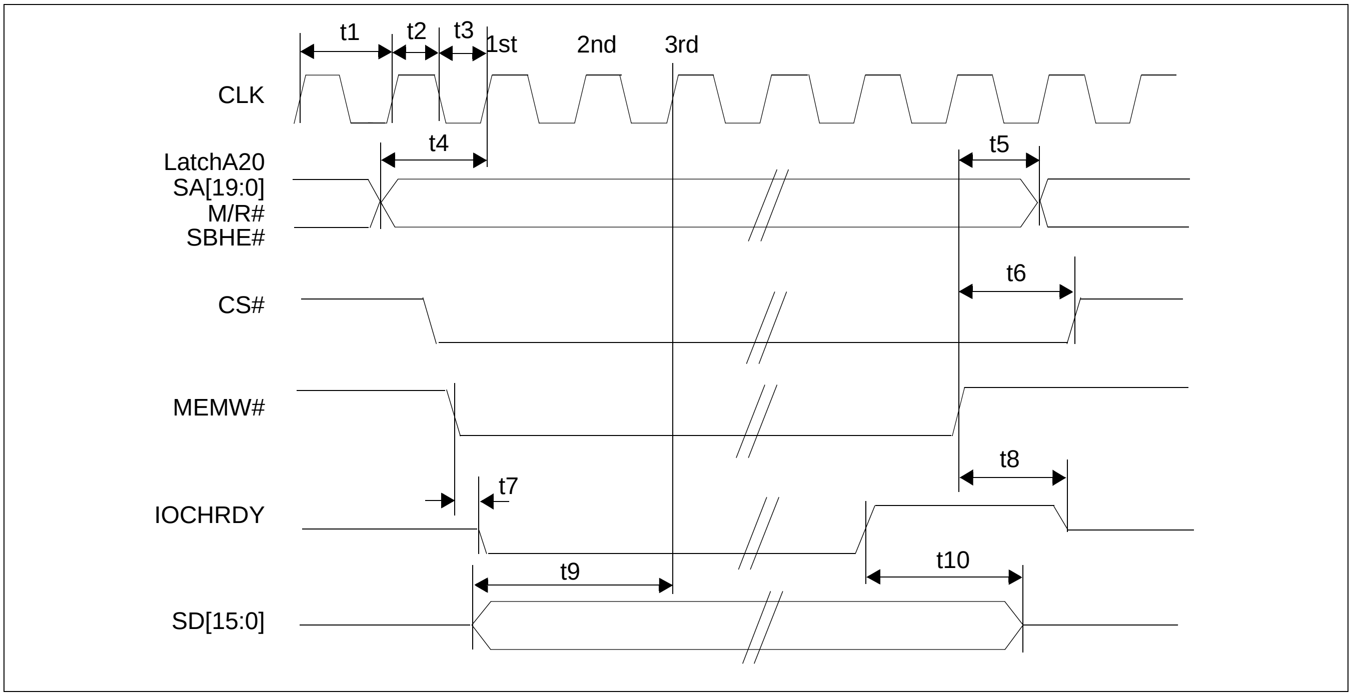


Figure 7-19: MIPS/ISA Host Interface Write Timing

Table 7-20 : MIPS/ISA Host Interface Write Timing

Symbol	Parameter	Min	Max	Units
f_{CLK}	Clock frequency	—	50	MHz
t1	Clock period	$1/f_{CLK}$	—	ns
t2	Clock pulse width high	7	—	ns
t3	Clock pulse width low	7	—	ns
t4	LatchA20, SA[19:0], M/R#, SBHE# setup to 1st CLK where CS# = 0 and MEMW# = 0	8	—	ns
t5	LatchA20, SA[19:0], M/R#, SBHE# hold from rising edge of MEMW#	0	—	ns
t6	CS# hold from rising edge of MEMW#	0	—	ns
t7	Falling edge of either MEMW# to IOCHRDY driven low	3	15	ns
t8	Rising edge of either MEMW# to IOCHRDY tristate	2	11	ns
t9	SD[15:0] setup to 3rd CLK where CS# = 0 MEMW# = 0	0	—	ns
t10	SD[15:0] hold	5	—	ns

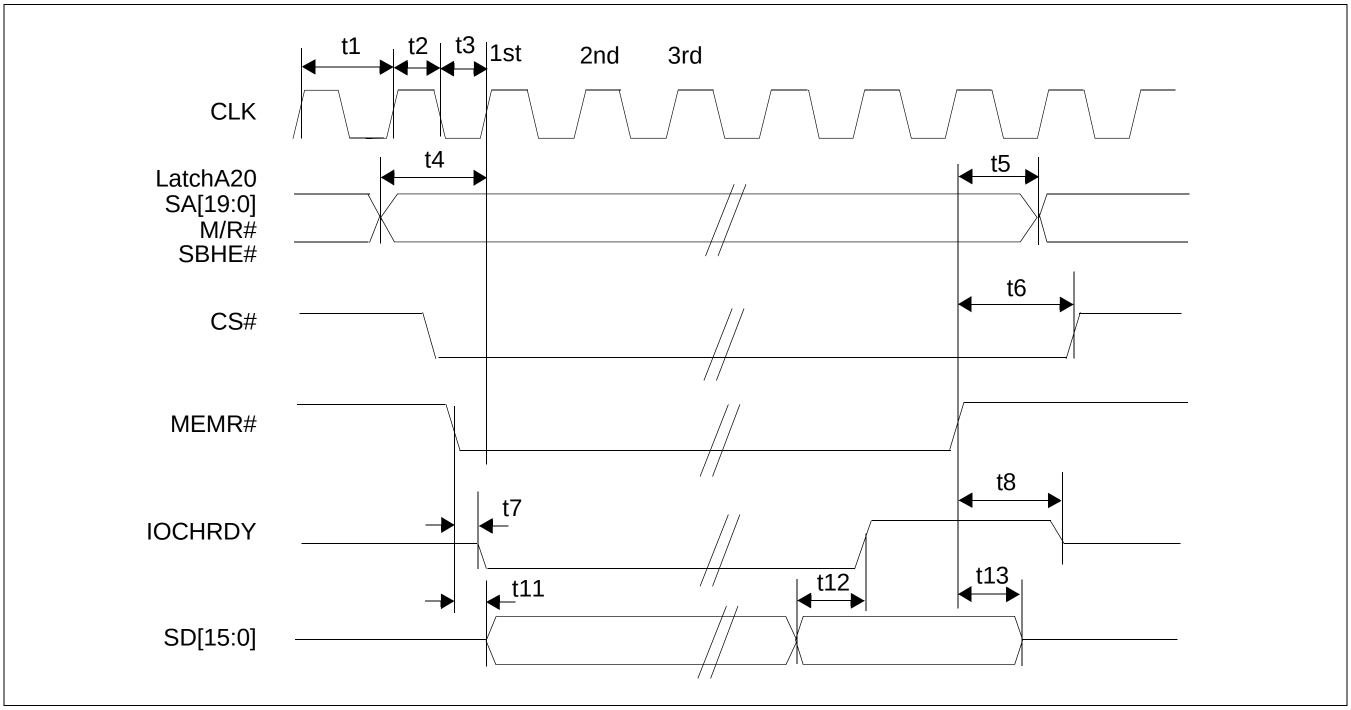


Figure 7-20: MIPS/ISA Host Interface Read Timing

Table 7-21 : MIPS/ISA Host Interface Read Timing

Symbol	Parameter	Min	Max	Units
f_{CLK}	Clock frequency	—	50	MHz
t1	Clock period	$1/f_{CLK}$	—	ns
t2	Clock pulse width high	7	—	ns
t3	Clock pulse width low	7	—	ns
t4	LatchA20, SA[19:0], M/R#, SBHE# setup to 1st CLK where CS# = 0 and MEMR# = 0	8	—	ns
t5	LatchA20, SA[19:0], M/R#, SBHE# hold from rising edge of MEMR#	0	—	ns
t6	CS# hold from rising edge of MEMR#	0	—	ns
t7	Falling edge of MEMR# to IOCHRDY driven low	3	15	ns
t8	Rising edge of MEMR# to IOCHRDY tristate	2	11	ns
t11	Falling edge MEMR# to SD[15:0] driven	4	—	ns
t12	SD[15:0] setup to rising edge IOCHRDY	0	—	ns
t13	Rising edge of MEMR# to SD[15:0] tristate	4	15	ns

7.4.8 Freescale MC68000

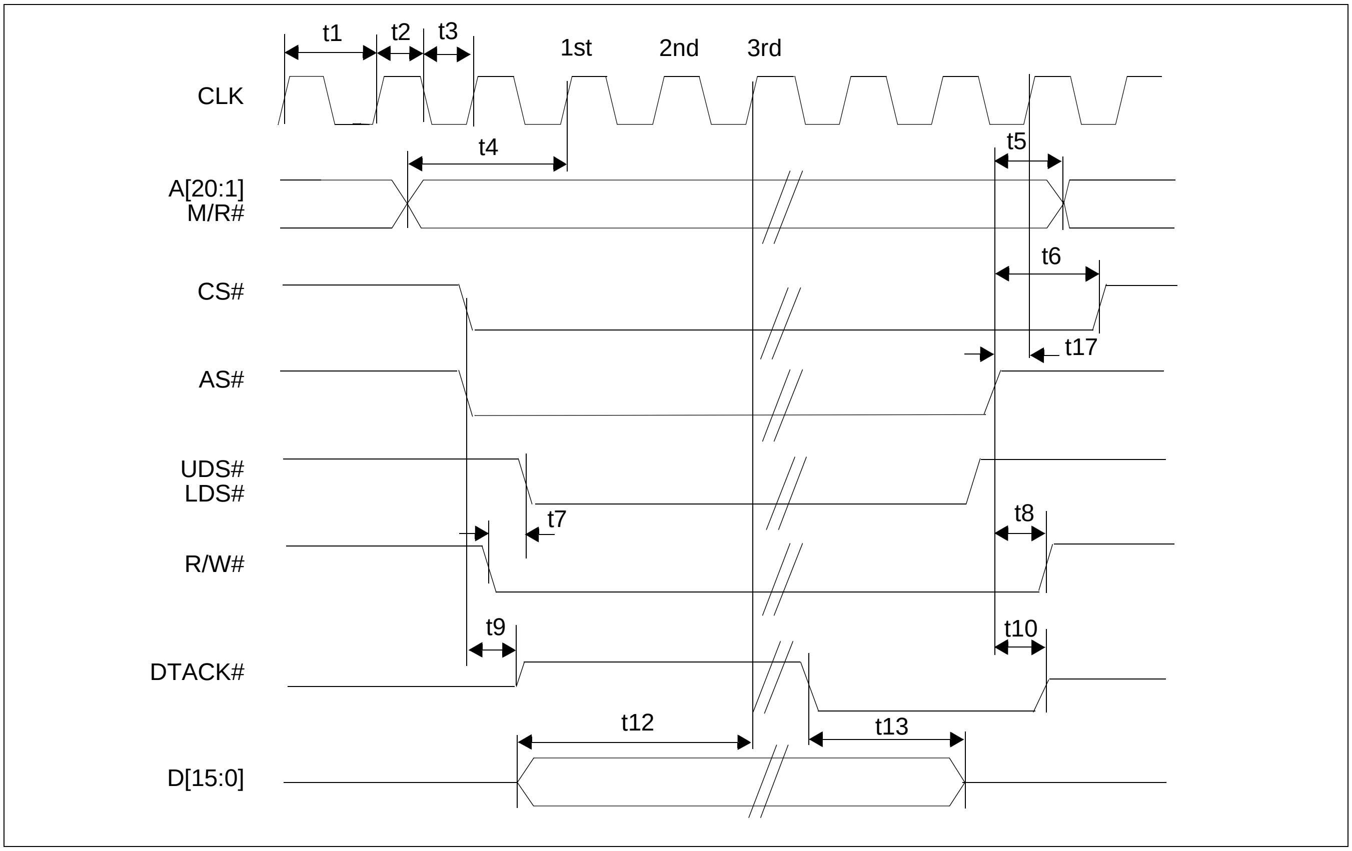


Figure 7-21: Freescale MC68000 Host Interface Write Timing

Table 7-22 : Freescale MC68000 Host Interface Write Timing

Symbol	Parameter	Min	Max	Units
f_{CLK}	Clock frequency	—	50	MHz
t1	Clock period	$1/f_{CLK}$	—	ns
t2	Clock pulse width high	6	—	ns
t3	Clock pulse width low	6	—	ns
t4	A[20:1], M/R# setup to 1st CLK where CS# = 0, AS# = 0, and either UDS# = 0 or LDS# = 0	7	—	ns
t5	A[20:1], M/R# hold from AS#	0	—	ns
t6	CS# hold from AS#	0	—	ns
t7	R/W# setup to either UDS# = 0 or LDS# = 0	10	—	ns
t8	R/W# hold from AS#	0	—	ns
t9	AS# = 0 and CS# = 0 to DTACK# driven high	1	—	ns
t10	AS# high to DTACK# high impedance	3	13	ns
t12	D[15:0] hold from 3rd CLK where CS# = 0, AS# = 0, and either UDS# = 0 or LDS# = 0	0	—	ns
t13	D[15:0] hold from falling edge of DTACK#	0	—	ns
t17	AS# high setup to CLK	6	—	ns

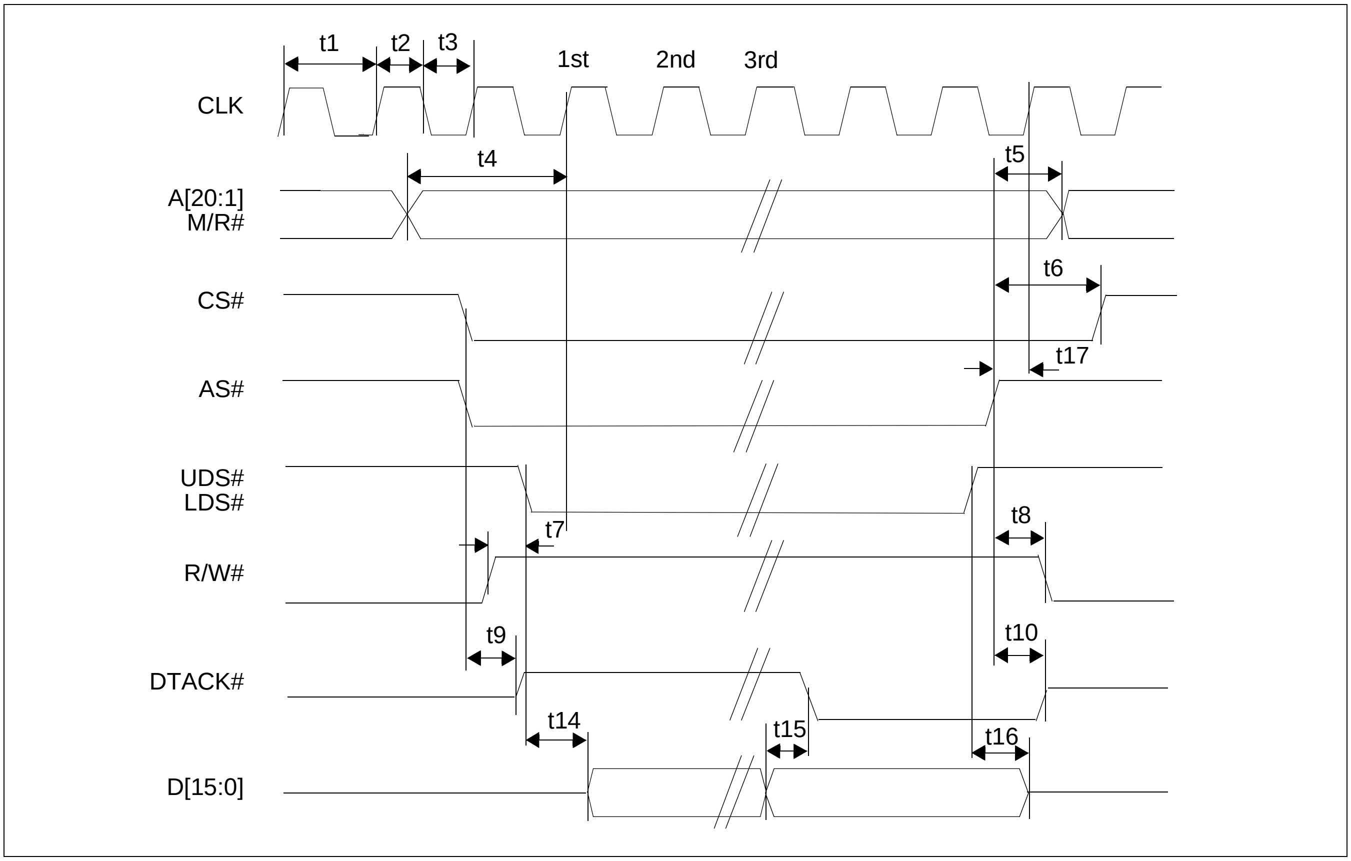


Figure 7-22: Freescale MC68000 Host Interface Read Timing

Table 7-23 : Freescale MC68000 Host Interface Read Timing

Symbol	Parameter	Min	Max	Units
f_{CLK}	Clock frequency	—	50	MHz
t1	Clock period	$1/f_{CLK}$	—	ns
t2	Clock pulse width high	6	—	ns
t3	Clock pulse width low	6	—	ns
t4	A[20:1], M/R# setup to 1st CLK where CS# = 0, AS# = 0, and either UDS# = 0 or LDS# = 0	7	—	ns
t5	A[20:1], M/R# hold from AS#	0	—	ns
t6	CS# hold from AS#	0	—	ns
t7	R/W# setup to either UDS# = 0 or LDS# = 0	10	—	ns
t8	R/W# hold from AS#	0	—	ns
t9	AS# = 0 and CS# = 0 to DTACK# driven high	1	—	ns
t10	AS# high to DTACK# high impedance	3	13	ns
t14	Falling edge of UDS# = 0 or LDS# = 0 to D[15:0] driven	3	—	ns
t15	D[15:0] valid to DTACK# falling edge	0	—	ns
t16	UDS# and LDS# high to D[15:0] invalid/high impedance	4	15	ns
t17	AS# high setup to CLK	6	—	ns

7.4.9 Freescale MC68030

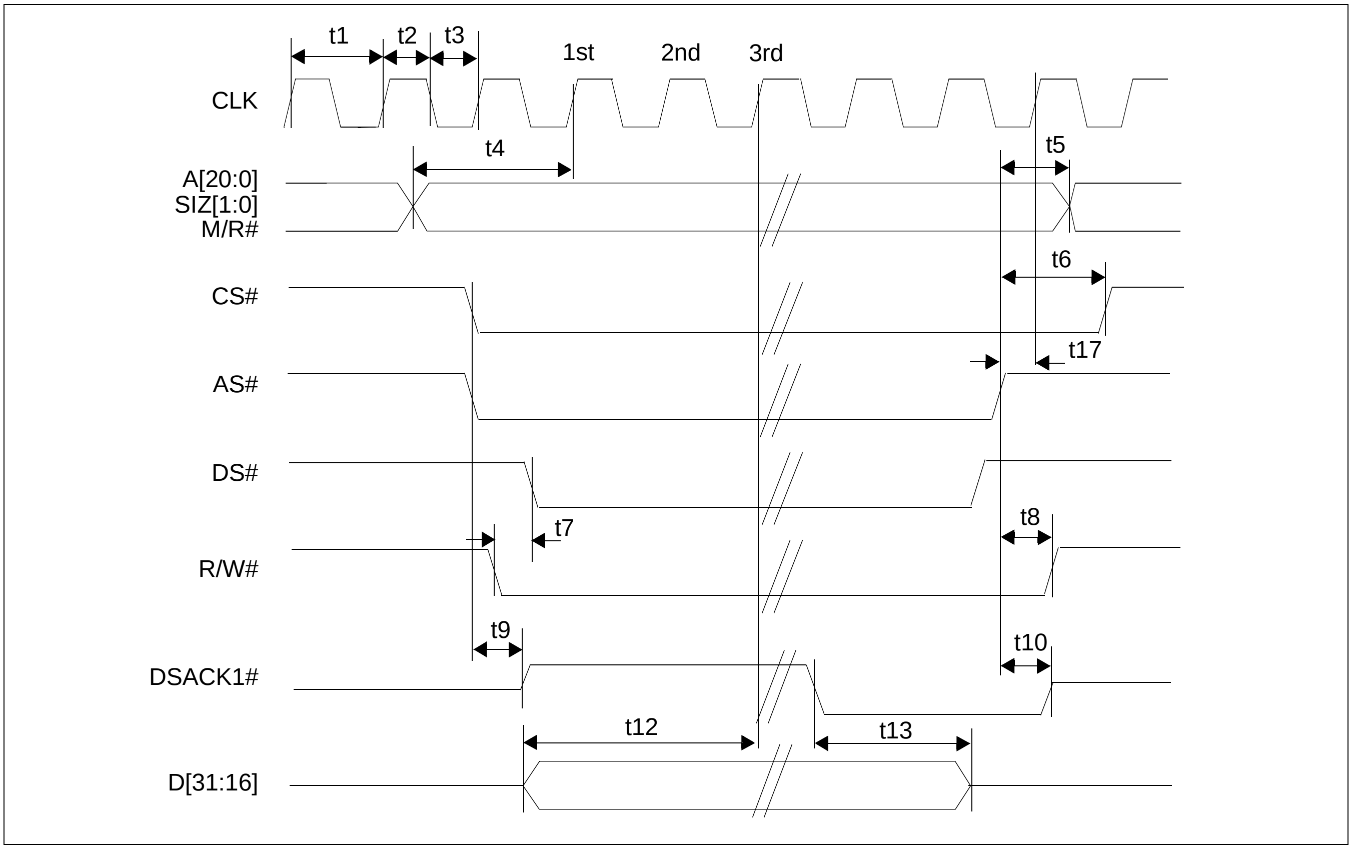


Figure 7-23: Freescale MC68030 Host Interface Write Timing

Table 7-24 : Freescale MC68030 Host Interface Write Timing

Symbol	Parameter	Min	Max	Units
f_{CLK}	Clock frequency	—	50	MHz
t1	Clock period	$1/f_{CLK}$	—	ns
t2	Clock pulse width high	6	—	ns
t3	Clock pulse width low	6	—	ns
t4	A[20:1], SIZ[1:0], M/R# setup to 1st CLK where CS# = 0 AS# = 0, DS# = 0	7	—	ns
t5	A[20:1], SIZ[1:0], M/R# hold from AS#	0	—	ns
t6	CS# hold from AS#	0	—	ns
t7	R/W# setup to DS#	10	—	ns
t8	R/W# hold from AS#	0	—	ns
t9	AS# = 0 and CS# = 0 to DSACK1# driven high	1	—	ns
t10	AS# high to DSACK1# high impedance	3	13	ns
t12	D[31:16] valid to 3rd CLK where CS# = 0 AS# = 0, and DS# = 0	0	—	ns
t13	D[31:16] hold from falling edge of DSACK1#	0	—	ns
t17	AS# high setup to CLK	6	—	ns

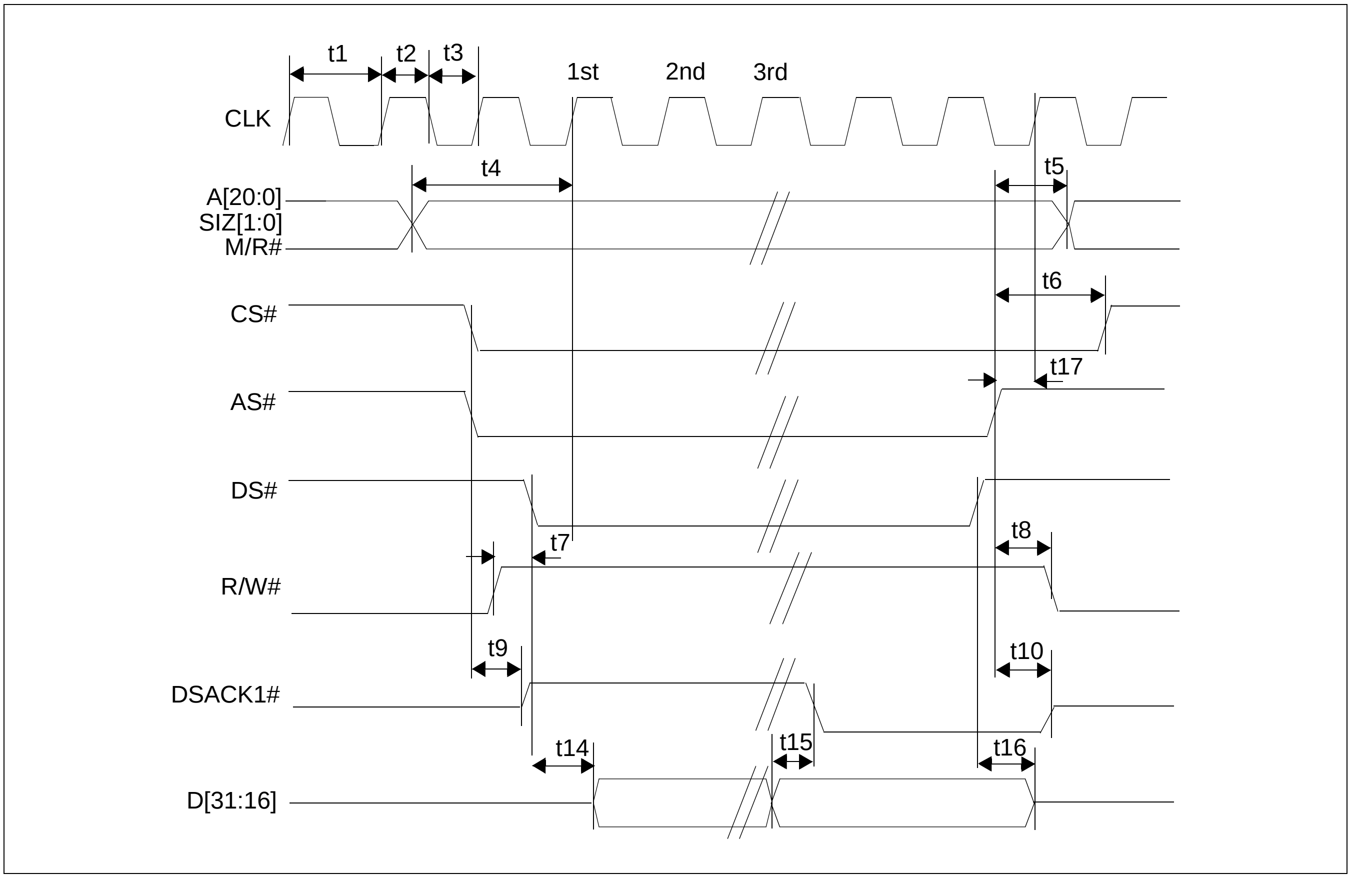


Figure 7-24: Freescale MC68030 Host Interface Read Timing

Table 7-25 : Freescale MC68030 Host Interface Read Timing

Symbol	Parameter	Min	Max	Units
f_{CLK}	Clock frequency	—	50	MHz
t1	Clock period	$1/f_{CLK}$	—	ns
t2	Clock pulse width high	6	—	ns
t3	Clock pulse width low	6	—	ns
t4	A[20:1], SIZ[1:0], M/R# setup to 1st CLK where CS# = 0 AS# = 0, DS# = 0	7	—	ns
t5	A[20:1], SIZ[1:0], M/R# hold from AS#	0	—	ns
t6	CS# hold from AS#	0	—	ns
t7	R/W# setup to DS#	10	—	ns
t8	R/W# hold from AS#	0	—	ns
t9	AS# = 0 and CS# = 0 to DSACK1# driven high	1	—	ns
t10	AS# high to DSACK1# high impedance	3	13	ns
t14	Falling edge of DS# = 0 to D[31:16] driven	3	—	ns
t15	D[31:16] valid to DSACK1# falling edge	0	—	ns
t16	DS# high to D[31:16] invalid/high impedance	4	15	ns
t17	AS# high setup to CLK	6	—	ns

7.4.10 Freescale MPC555 (Non-burst Mode)

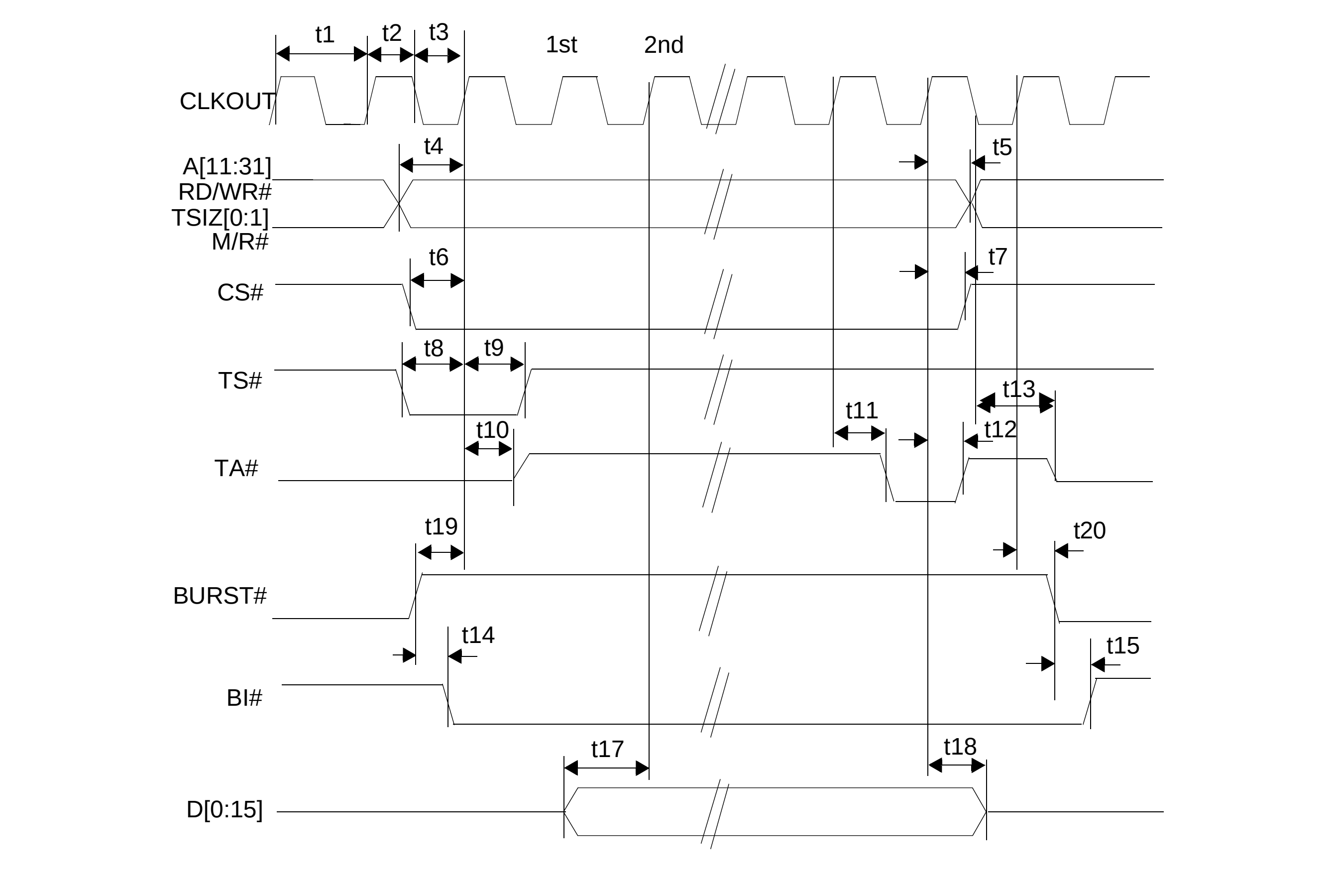


Figure 7-25: Freescale MPC555 Host Interface Write Timing (Non-burst Mode)

Table 7-26 : Freescale MPC555 Host Interface Write Timing (Non-burst Mode)

Symbol	**Parameter**	**Min**	**Max**	**Units**
f_{CLKOUT}	Clock frequency	—	40	MHz
t1	Clock period	$1/f_{CLKOUT}$	—	ns
t2	Clock pulse width high	6	—	ns
t3	Clock pulse width low	6	—	ns
t4	AB[11:31], RD/WR#, TSIZ[0:1], M/R# setup	3	—	ns
t5	AB[11:31], RD/WR#, TSIZ[0:1], M/R# hold	0	—	ns
t6	CS# setup	0	—	ns
t7	CS# hold	3	—	ns
t8	TS# setup	1	—	ns
t9	TS# hold	2	—	ns
t10	CLKOUT to TA# driven	4	—	ns
t11	CLKOUT to TA# low	4	13	ns
t12	CLKOUT to TA# high	5	14	ns
t13	Negative edge CLKOUT to TA# tristate	3	12	ns
t14	BURST# high to BI# low	5	15	ns
t15	BURST# low to BI# high	3	12	ns
t17	DB[15:0] setup to 2nd CLKOUT after TS# = 0	0	—	ns
t18	CLKOUT to DB[15:0] hold	2	—	ns
t19	BURST# setup	5	—	ns
t20	BURST# hold	3	—	ns

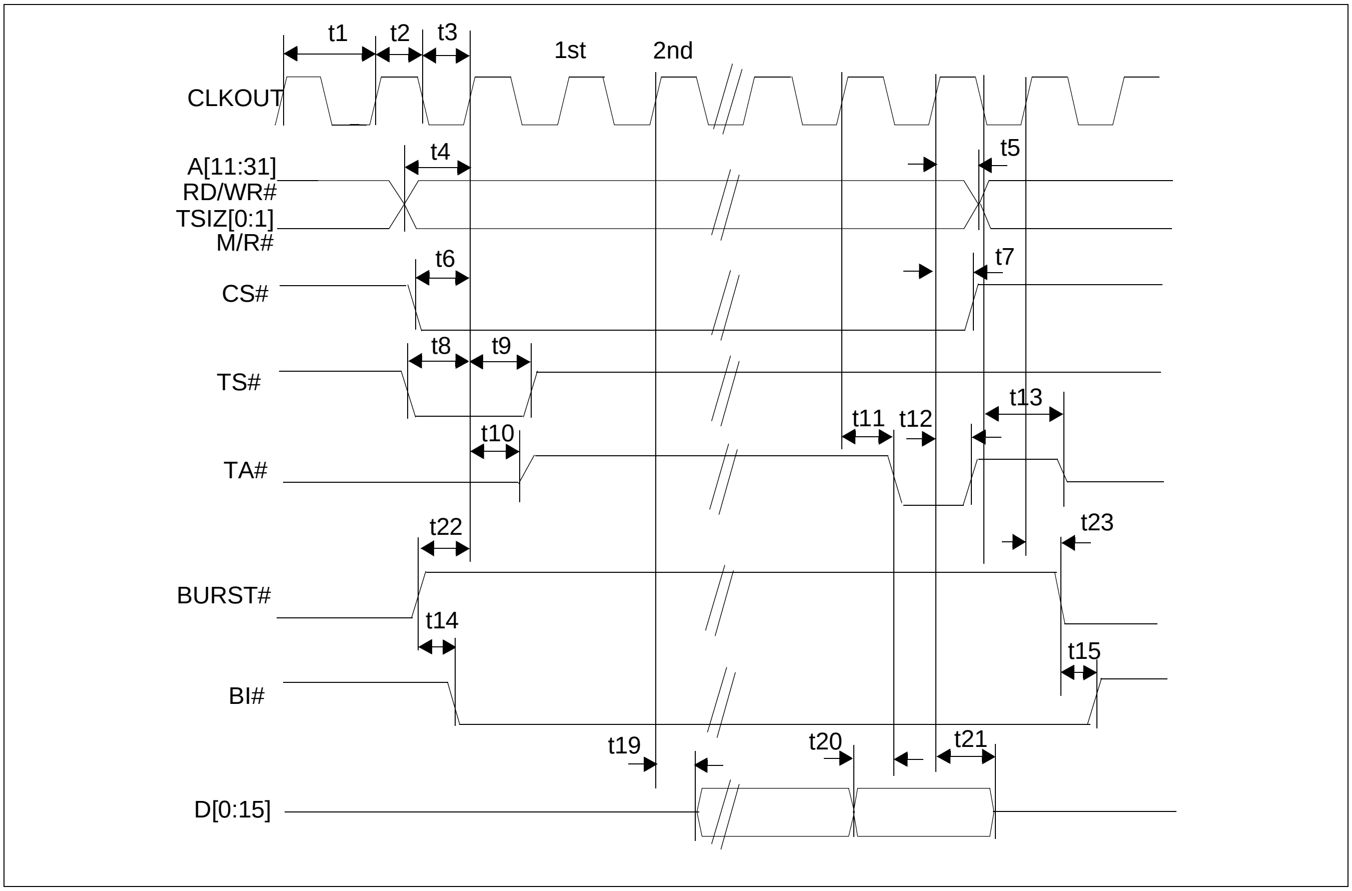


Figure 7-26: Freescale MPC555 Host Interface Read Timing (Non-burst Mode)

Table 7-27 : Freescale MPC555 Host Interface Read Timing (Non-burst Mode)

Symbol	**Parameter**	**Min**	**Max**	**Units**
f_{CLKOUT}	Clock frequency	—	40	MHz
t1	Clock period	$1/f_{CLKOUT}$	—	ns
t2	Clock pulse width high	6	—	ns
t3	Clock pulse width low	6	—	ns
t4	AB[11:31], RD/WR#, TSIZ[0:1], M/R# setup	3	—	ns
t5	AB[11:31], RD/WR#, TSIZ[0:1], M/R# hold	0	—	ns
t6	CS# setup	0	—	ns
t7	CS# hold	3	—	ns
t8	TS# setup	1	—	ns
t9	TS# hold	2	—	ns
t10	CLKOUT to TA# driven	4	—	ns
t11	CLKOUT to TA# low	4	13	ns
t12	CLKOUT to TA# high	5	14	ns
t13	negative edge CLKOUT to TA# tristate	3	12	ns
t14	BURST# high to BI# low	5	15	ns
t15	BURST# low to BI# high	3	12	ns
t19	CLKOUT to DB driven	-20	—	ns
t20	DB[15:0] valid to TA# falling edge	0	—	ns
t21	CLKOUT to DB[15:0] tristate	3	12	ns
t22	BURST# setup	5	—	ns
t23	BURST# hold	3	—	ns

7.4.11 Philips PR31500/PR31700 / Toshiba TX3912

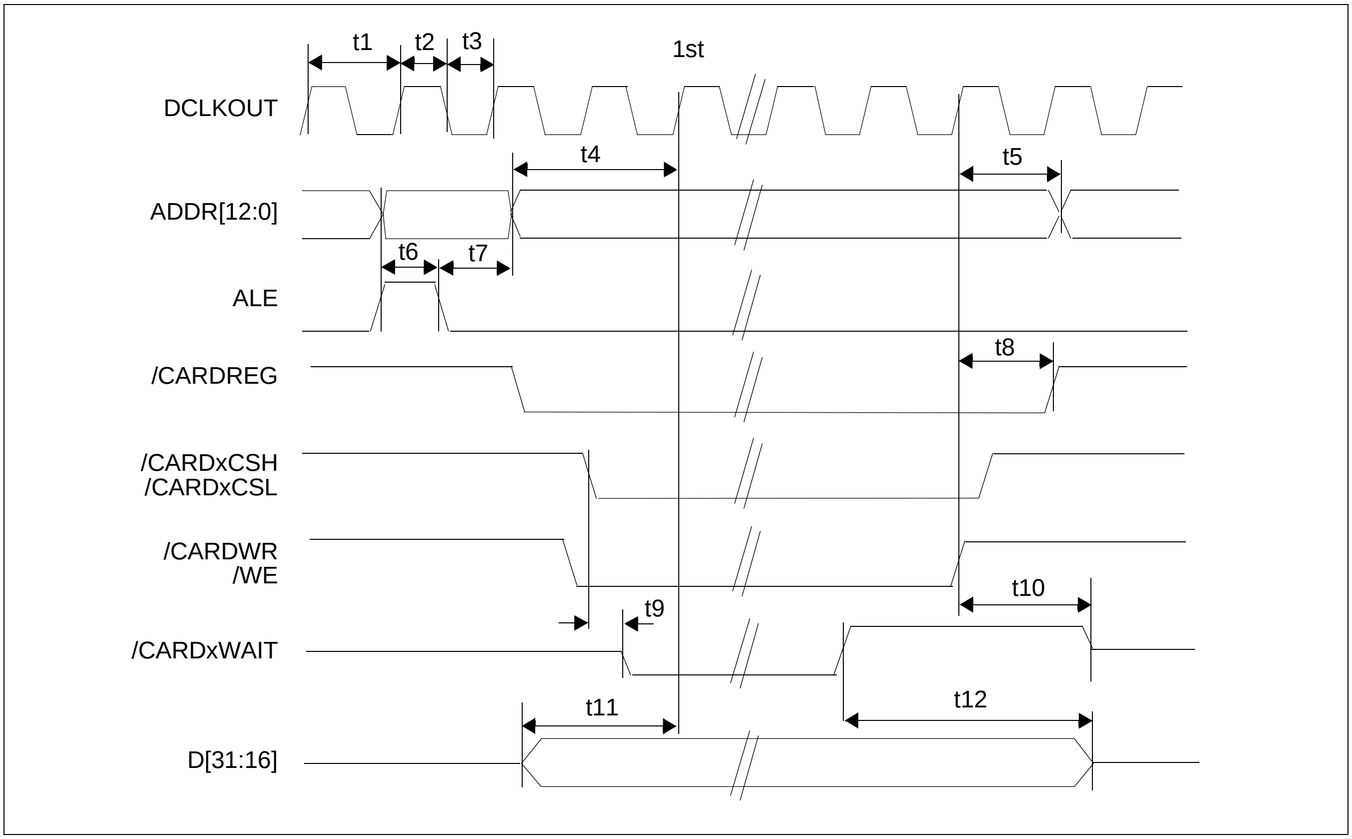


Figure 7-27: Philips PR31500/PR31700 / Toshiba TX3912 Host Interface Write Timing

Note

For the Toshiba TX3912, active low signals are designated using an “*” after the signal name, instead of a “/” before the signal name. For example, the PR31500 signal /CARDREG is the same as the TX3912 signal CARDREG*.

Table 7-28 : Philips PR31500/PR31700 / Toshiba TX3912 Host Interface Write Timing

Symbol	Parameter	Min	Max	Units
$f_{DCLKOUT}$	Clock frequency	—	50	MHz
t1	Clock period	$1/f_{DCLKOUT}$	—	ns
t2	Clock pulse width high	6	—	ns
t3	Clock pulse width low	6	—	ns
t4	ADDR[12:0] setup to 1st CLK of cycle	10	—	ns
t5	ADDR[12:0] hold from command invalid	5	—	ns
t6	ADDR[12:0] setup to falling edge ALE	10	—	ns
t7	ADDR[12:0] hold from falling edge ALE	5	—	ns
t8	/CARDREG hold from command invalid	0	—	ns
t9	Falling edge of chip select to /CARDxWAIT driven	0	15	ns
t10	Command invalid to /CARDxWAIT tristate	3	15	ns
t11	D[31:16] valid to 1st CLK of cycle	10	—	ns
t12	D[31:16] hold from rising edge of /CARDxWAIT	0	—	ns

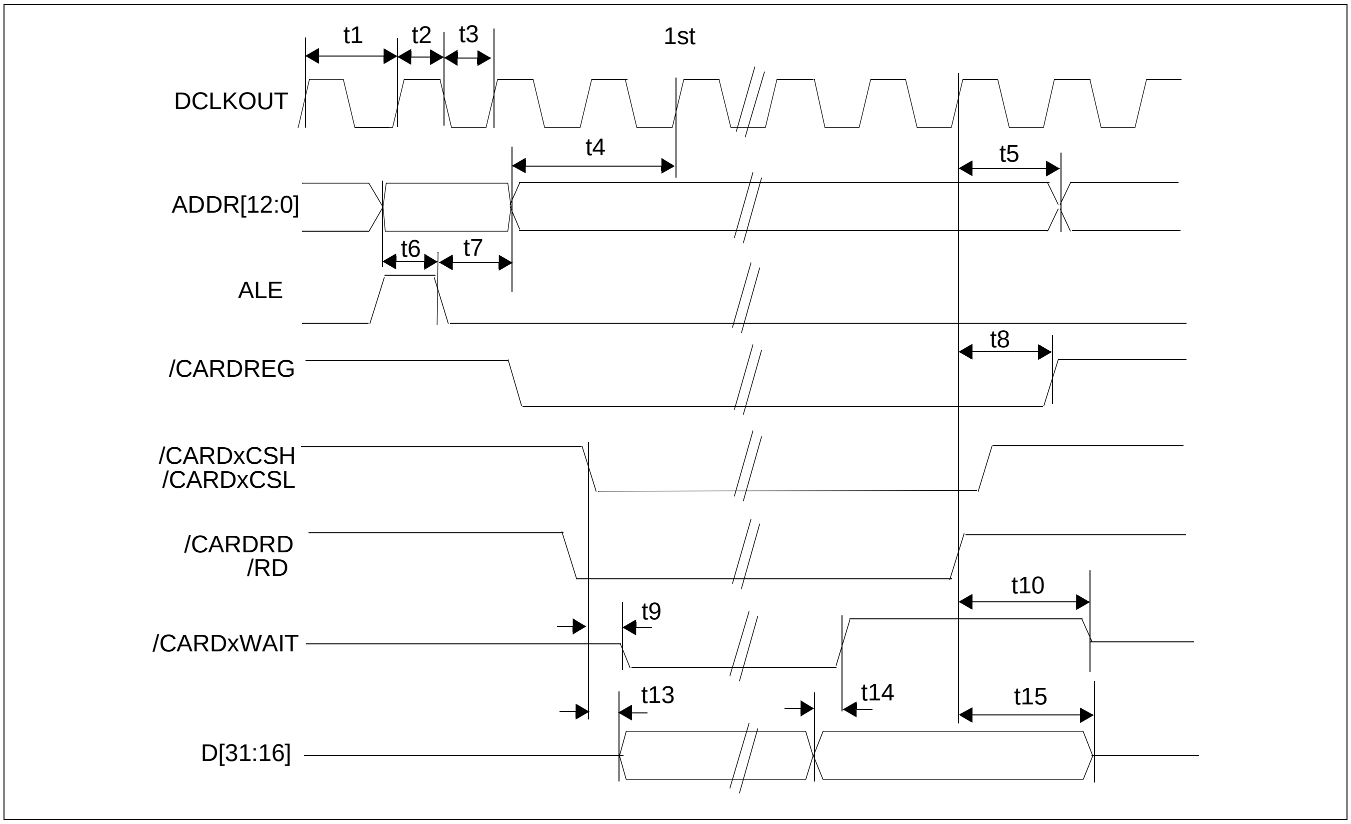


Figure 7-28: Philips PR31500/PR31700 / Toshiba TX3912 Host Interface Read Timing

Note

For the Toshiba TX3912, active low signals are designated using an "*" after the signal name, instead of a "/" before the signal name. For example, the PR31500 signal /CARDREG is the same as the TX3912 signal CARDREG*.

Table 7-29 : Philips PR31500/PR31700 / Toshiba TX3912 Host Interface Read Timing

Symbol	Parameter	Min	Max	Units
$f_{DCLKOUT}$	Clock frequency	—	50	MHz
t1	Clock period	$1/f_{DCLKOUT}$	—	ns
t2	Clock pulse width high	6	—	ns
t3	Clock pulse width low	6	—	ns
t4	ADDR[12:0] setup to 1st CLK of cycle	10	—	ns
t5	ADDR[12:0] hold from command invalid	5	—	ns
t6	ADDR[12:0] setup to falling edge ALE	10	—	ns
t7	ADDR[12:0] hold from falling edge ALE	5	—	ns
t8	/CARDREG hold from command invalid	0	—	ns
t9	Falling edge of chip select to /CARDxWAIT driven	0	15	ns
t10	Command invalid to /CARDxWAIT tristate	3	15	ns
t13	Chip select to D[31:16] driven	1	—	ns
t14	D[31:16] setup to rising edge /CARDxWAIT	0	—	ns
t15	Command invalid to D[31:16] tristate	4	15	ns

7.4.12 Serial Host

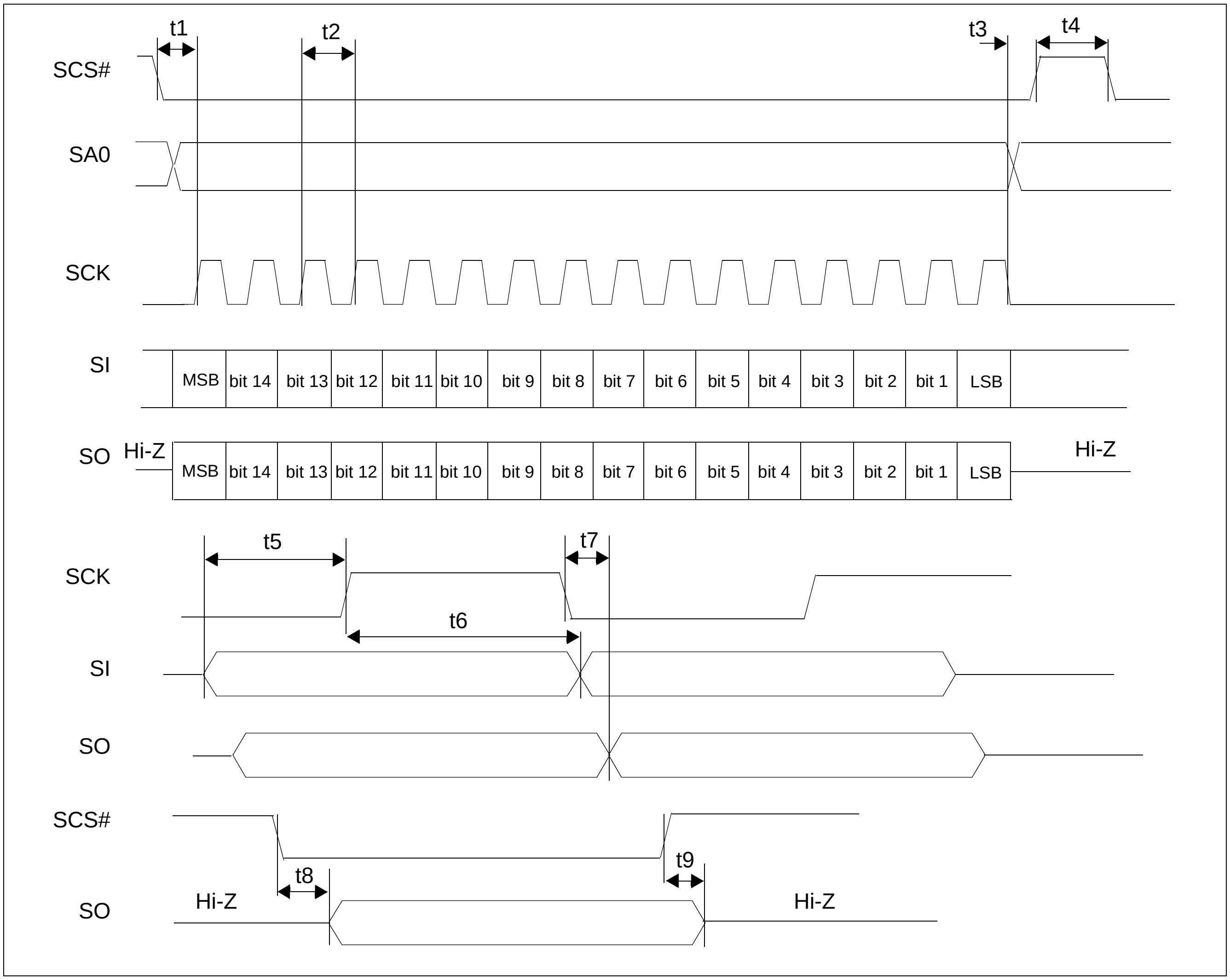


Figure 7-29: Serial Host Interface Timing for Data Valid on Falling Edge of SCK

Table 7-30 : Serial Host Interface Timing for Data Valid on Falling Edge of SCK

Symbol	Parameter	Min	Max	Units
t1	SCS# low, SA0 active to rising edge of SCK	2	—	ns
t2	SCK period	63	—	ns
t3	Falling edge of SCK to SCS# high	5	—	ns
t4	SCS# high pulse width	1	—	SCK
t5	SI data setup time to rising edge of SCK	5	—	ns
t6	SI data hold time from rising edge of SCK	5	—	ns
t7	SO data valid from falling edge of SCK	—	14	ns
t8	Falling edge of SCS# to SO active	—	13	ns
t9	Rising edge of SCS# to SO Hi-Z	—	11	ns

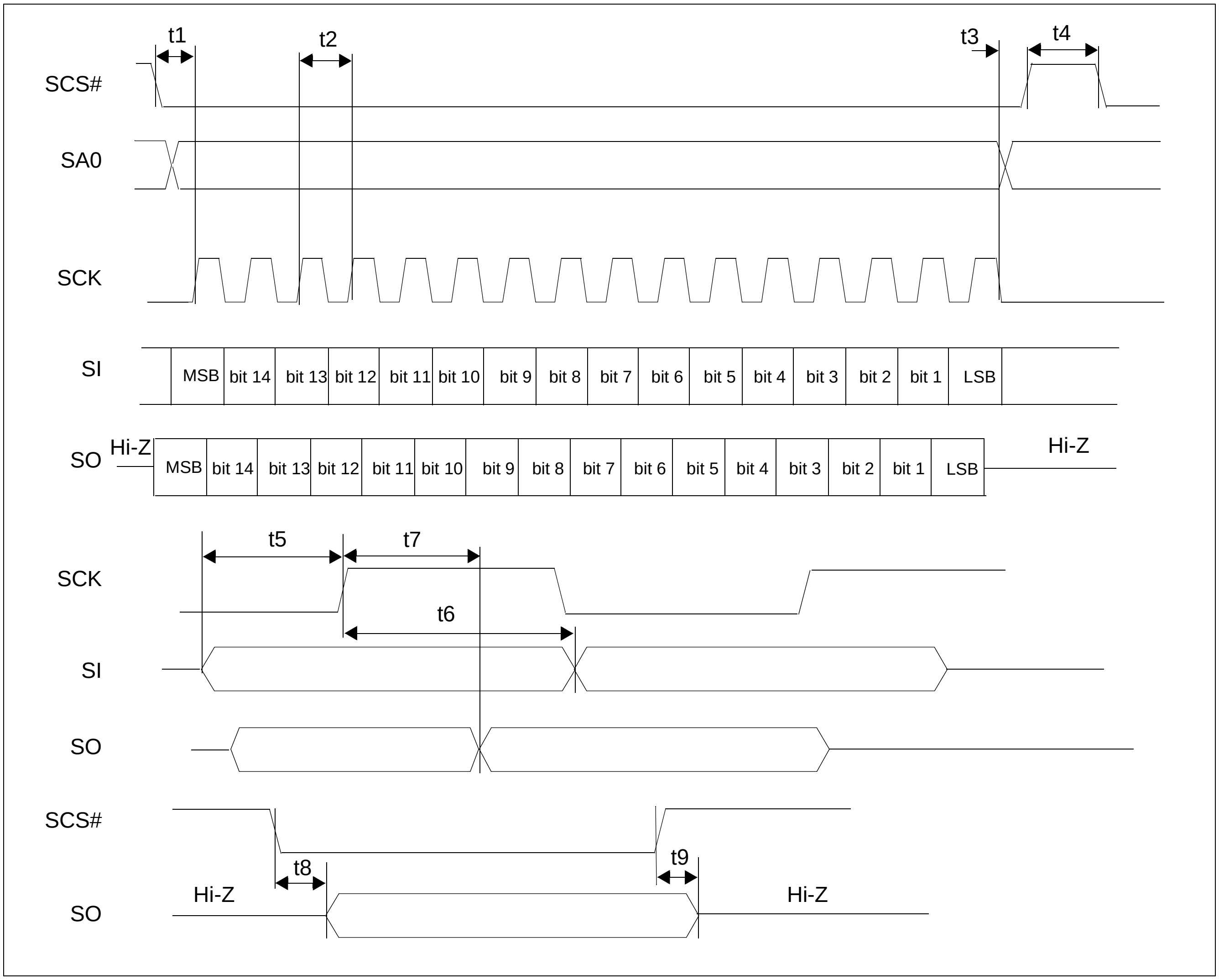


Figure 7-30: Serial Host Interface Timing for Data Valid on Rising Edge of SCK

Table 7-31 : Serial Host Interface Timing for Data Valid on Rising Edge of SCK

Symbol	Parameter	Min	Max	Units
t1	SCS# low, SA0 active to rising edge of SCK	2	—	ns
t2	SCK period	63	—	ns
t3	Falling edge of SCK to SCS# high	5	—	ns
t4	SCS# high pulse width	1	—	SCK
t5	SI data setup time to rising edge of SCK	5	—	ns
t6	SI data hold time from rising edge of SCK	5	—	ns
t7	SO data valid from rising edge of SCK	—	14	ns
t8	Falling edge of SCS# to SO active	—	13	ns
t9	Rising edge of SCS# to SO Hi-Z	—	11	ns

7.4.13 Serial Host Interface Burst Mode

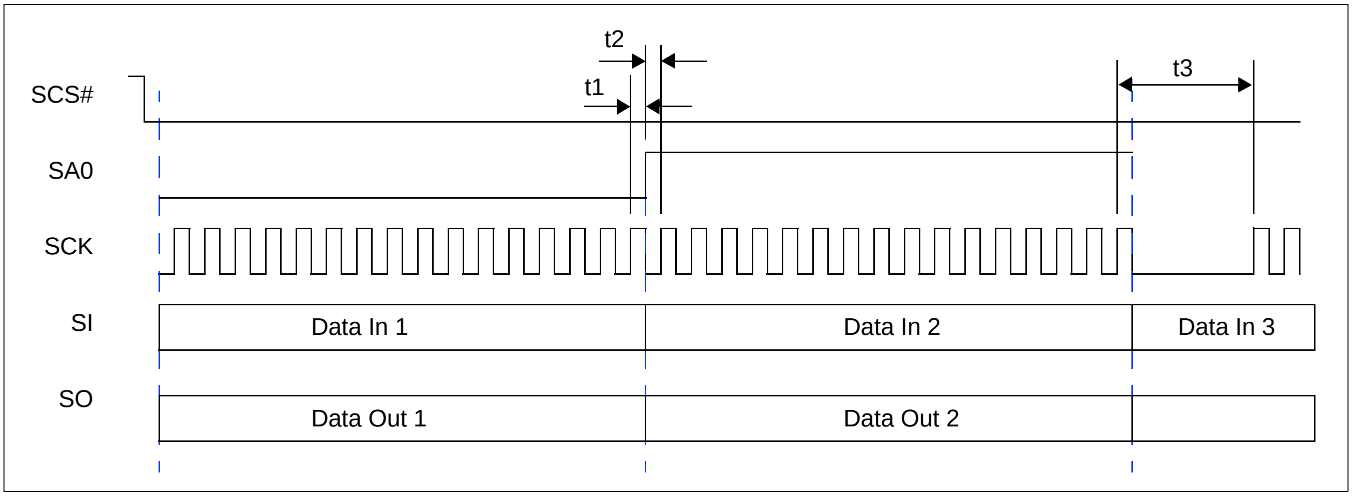


Figure 7-31: Serial Host Interface Burst Mode Timing

Table 7-32 : Serial Host Interface Burst Mode Timing

Symbol	Parameter	Min	Max	Units
t1	SA0 hold after the rising edge of SCK	3	SCK-t2	ns
t2	SA0 setup to the rising edge of SCK	1	—	ns
t3	Time between the rising edge of SCK of the last transfer and the rising edge of the next transfer	5	—	ns

7.5 Power Sequencing

Setting REG[0470h] bit 0 = 1b places the S1D13513 into power save mode. Internally the only clock on is Host Bus interface. All other clocks are off.

7.6 Panel Interface Timing

7.6.1 Generic TFT Panel Timing

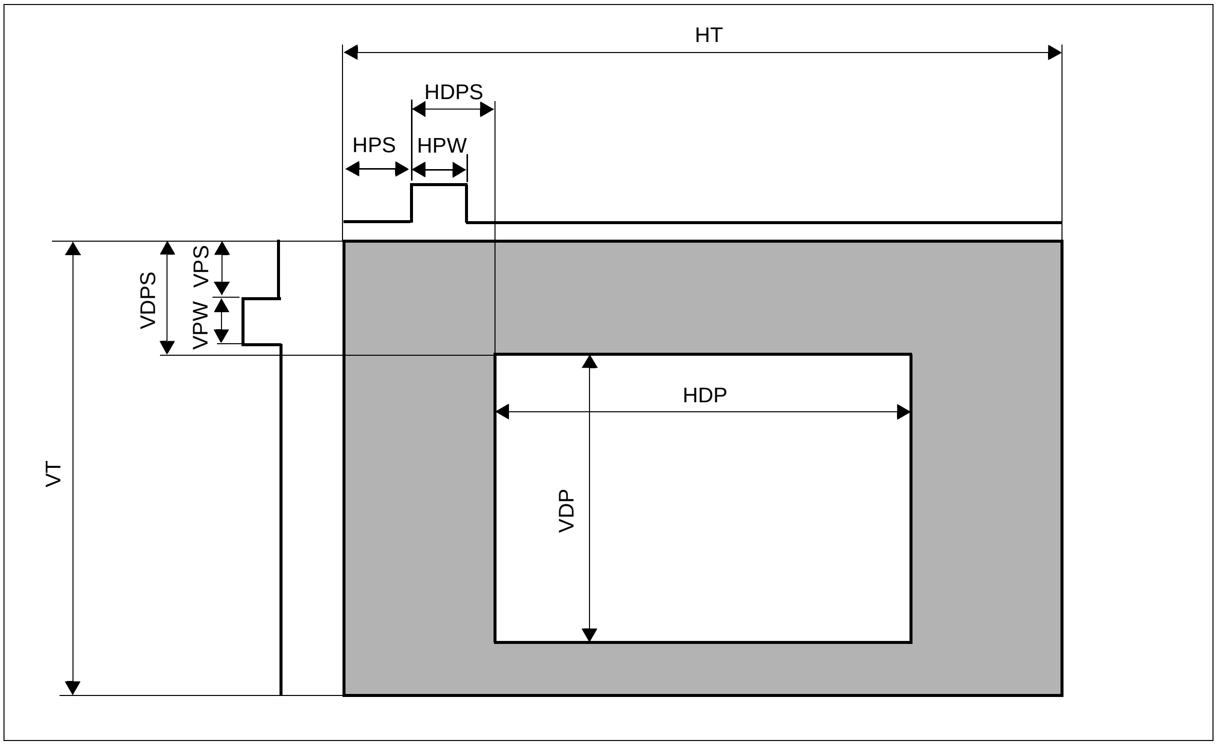


Figure 7-32: Generic TFT Panel Timing

Table 7-33: Generic TFT Panel Timing

Symbol	Description	Derived From	Units
HT	Horizontal Total (FPLINE period)	REG[0802h] bits 11-0 + 1	Ts
HDP	Horizontal Display Period	(REG[0804h] bits 10-0 + 1) x 2	
HDPS	Horizontal Display Period Start Position	REG[0806h] bits 11-0 +1	
HPW	Horizontal Pulse (FPLINE) Width	REG[0808h] bits 8-0 + 1	
HPS	Horizontal Pulse (FPLINE) Start Position	REG[080Ah] bits 11-0	
VT	Vertical Total (FPFRAME period)	REG[080Ch] bits 11-0 + 1	Lines
VDP	Vertical Display Period	REG[080Eh] bits 11-0 + 1	
VDPS	Vertical Display Period Start Position	REG[0810h] bits 11-0	
VPW	Vertical Pulse (FPFRAME) Width	REG[0812h] bits 4-0 + 1	
VPS	Vertical Pulse (FPFRAME) Start Position	REG[0814h] bits 11-0	

1. The following formulas must be valid for all panel timings:
   $HDPS + HDP < HT$
   $VDPS + VDP < VT$

Generic RGB Type Interface Panel Horizontal Timing

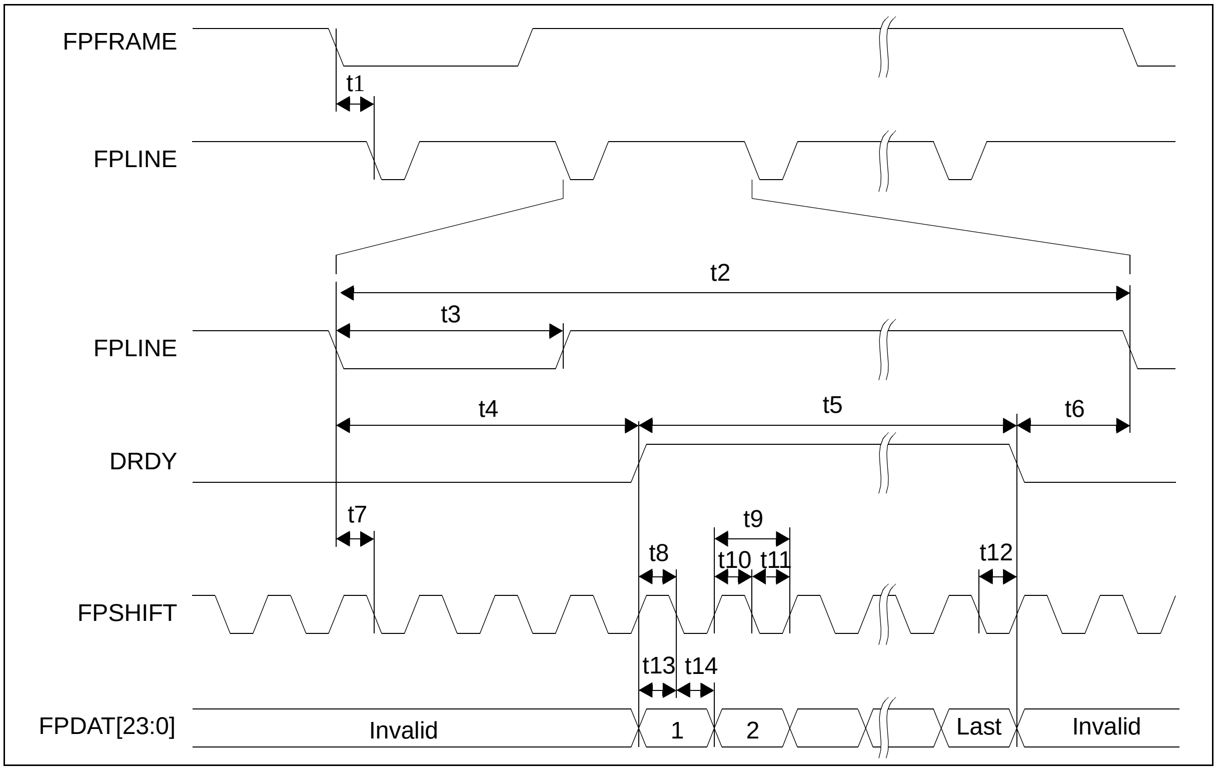


Figure 7-33: Generic RGB Type Interface Panel Horizontal Timing

Table 7-34: Generic RGB Type Interface Panel Horizontal Timing

Symbol	Parameter	Min	Typ	Max	Units
t1	FPFRAME falling edge to FPLINE falling edge	—	HPS	—	Ts (Note 1)
t2	Horizontal total period	—	HT	—	Ts
t3	FPLINE pulse width	—	HPW	—	Ts
t4	FPLINE falling edge to DRDY active	—	HDPS	—	Ts
t5	Horizontal display period	—	HDP	—	Ts
t6	DRDY falling edge to FPLINE falling edge	—	Note 2	—	Ts
t7	FPLINE setup time to FPSHIFT falling edge	0.5Ts - 1	0.5Ts	—	ns
t8	DRDY setup to FPSHIFT falling edge	0.5Ts	—	—	ns
t9	FPSHIFT period	—	1Ts	—	ns
t10	FPSHIFT pulse width high	—	0.5Ts	—	ns
t11	FPSHIFT pulse width low	—	0.5Ts	—	ns
t12	DRDY hold from FPSHIFT falling edge	0.5Ts - 3	0.5Ts	—	ns
t13	Data setup to FPSHIFT falling edge	0.5Ts - 1	0.5Ts	—	ns
t14	Data hold from FPSHIFT falling edge	0.5Ts - 3	0.5Ts	—	ns

1. Ts = pixel clock period
2. t6typ = t2 - t4 - t5
3. The Generic TFT timings are based on the following:
   FPSHIFT Pulse Polarity is 1b (REG[0800h] bit 7 = 1b) so all panel interface signals change at the rising edge of FPSHIFT.
   FPLINE Pulse Polarity bit is active low (REG[0808h] bit 15 = 0b).
   FPFRAME Pulse Polarity bit is active low (REG[0812h] bit 15 = 0b).

Generic RGB Type Interface Panel Vertical Timing

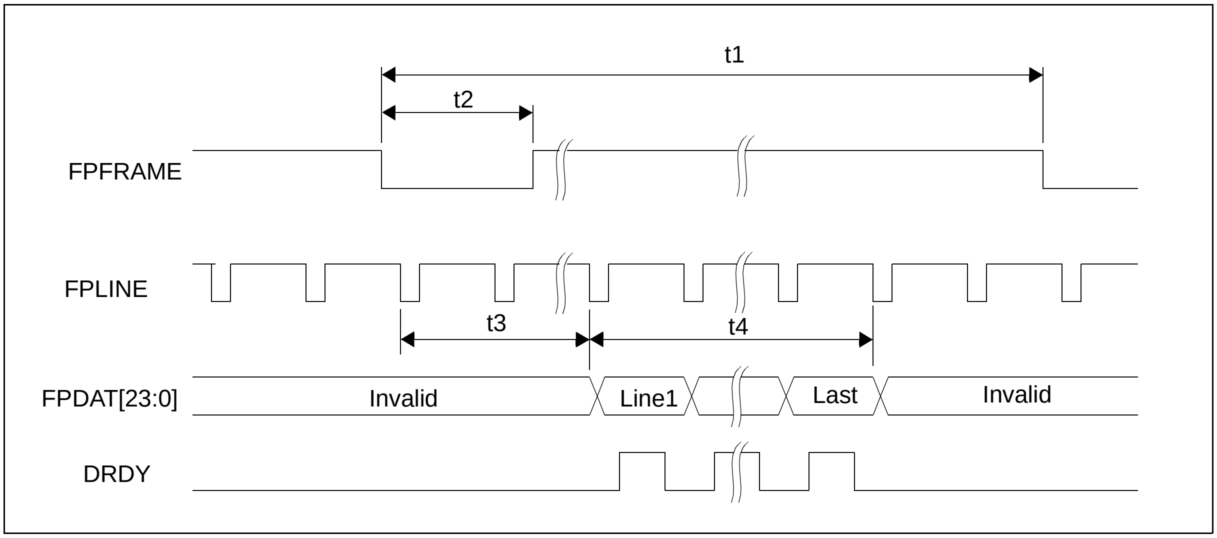


Figure 7-34: Generic RGB Type Interface Panel Vertical timing

Table 7-35: Generic RGB Type Interface Panel Vertical Timing

Symbol	Parameter	Min	Typ	Max	Units
t1	Vertical total period	—	VT	—	Lines
t2	FPFRAME pulse width	—	VPW	—	Lines
t3	Vertical display start position (Note 1)	—	Note 2	—	Lines
t4	Vertical display period	—	VDP	—	Lines

1. t3 is measured from the first FPLINE pulse at the start of the frame to the last FPLINE pulse before FPDAT is valid.
2. t3typ = VDPS - VPS

7.6.2 HR-TFT Panel Timing

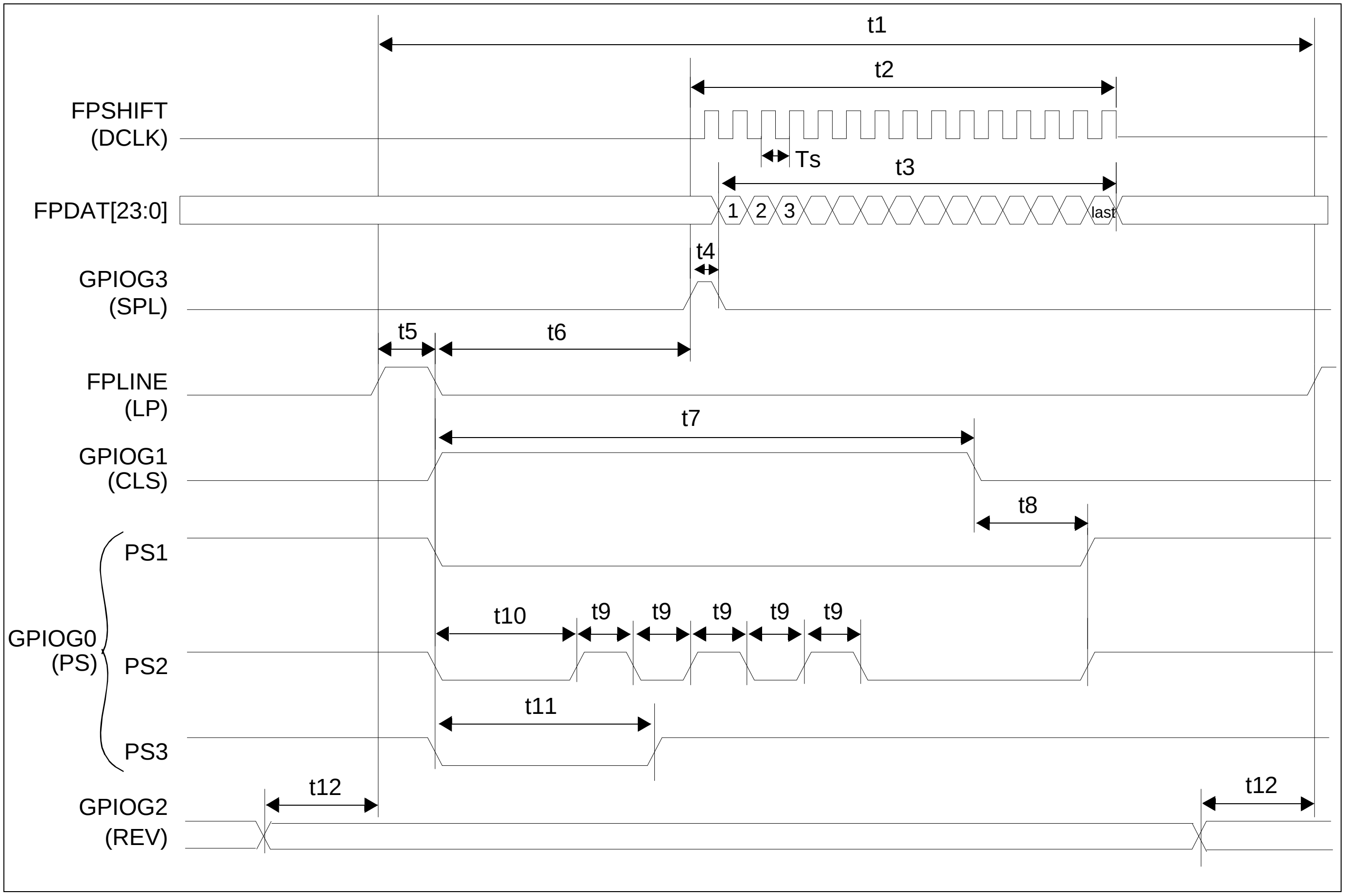


Figure 7-35: HR-TFT Panel Horizontal Timing

Table 7-36: HR-TFT Panel Horizontal Timing

Symbol	Parameter	Min	Typ	Max	Units
t1	Horizontal total period	—	Note 2	—	Ts (Note 1)
t2	FPSHIFT active	—	Note 3	—	Ts
t3	Horizontal display period	—	Note 4	—	Ts
t4	SPL pulse width	—	1	—	Ts
t5	FPLINE pulse width	—	Note 5	—	Ts
t6	FPLINE falling edge to SPL rising edge	—	Note 6	—	Ts
t7	CLS pulse width	—	Note 7	—	Ts
t8	CLS falling edge to GPIO0 (PS1) rising edge	—	Note 8	—	Ts
t9	PS2 toggle width	—	Note 9	—	Ts
t10	PS2 first falling edge to GPIO0 (PS2) first rising edge	—	Note 10	—	Ts
t11	PS3 pulse width	—	Note 11	—	Ts
t12	REV toggle position to FPLINE rising edge	—	Note 12	—	Ts

1. Ts = pixel clock period
2. t1typ = (REG[0802h] bits 11-0) + 1
3. t2typ = [((REG[0804h] bits 10-0) + 1) x 2] + 1
4. t3typ = [(REG[0804h] bits 10-0) + 1] x 2
5. t5typ = (REG[0808h] bits 8-0) + 1
6. t6typ = REG[0806h] bits 11-0 - REG[0808h] bits 8-0 + 2
7. t7typ = (REG[0822h] bits 10-0) > 0
8. t8typ = (REG[0824h] bits 7-0)
9. t9typ = (REG[0828h] bits 8-0) > 0
10. t10typ = (REG[0826h] bits 9-0) > 0
11. t11typ = (REG[082Ah] bits 8-0) > 0
12. t12typ = REG[082Ch] bits 6-0
13. The HR-TFT timings are based on the following:
    FPSHIFT Pulse Polarity is 0b (REG[0800h] bit 7 = 0b) so all panel interface signals change at the falling edge of FPSHIFT.

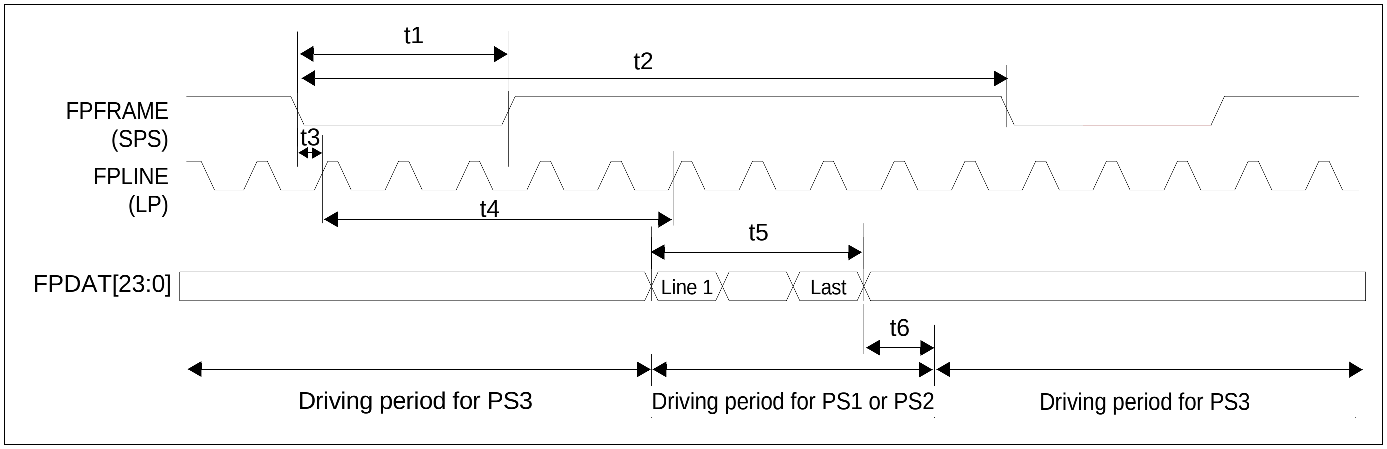


Figure 7-36: HR-TFT Panel Vertical Timing

Table 7-37: HR-TFT Panel Vertical Timing

Symbol	Parameter	Min	Typ	Max	Units
t1	FPFRAME pulse width	—	Note 2	—	Lines
t2	Vertical total period	—	Note 3	—	Lines
t3	FPFRAME rising/falling edge to FPLINE rising edge	—	1 (Note 4)	—	Ts (Note 1)
t4	Vertical display start position	—	Note 5	—	Lines
t5	Vertical display period	—	Note 6	—	Lines
t6	Extra driving period for PS1/2	—	Note 7	—	Lines

1. Ts = pixel clock period
2. t1typ = (REG[0812h] bits 4-0) + 1
3. t2typ = (REG[080Ch] bits 11-0) + 1
4. t3typ = (REG[080Ah] bits 11-0)
5. t4typ = REG[0810h] bits 11-0 - REG[0814h] bits 11-0
6. t5typ = (REG[080Eh] bits 11-0) + 1
7. t6typ = (REG[082Eh] bits 4-0)

7.6.3 ND-TFD 8-Bit Serial Interface Timing

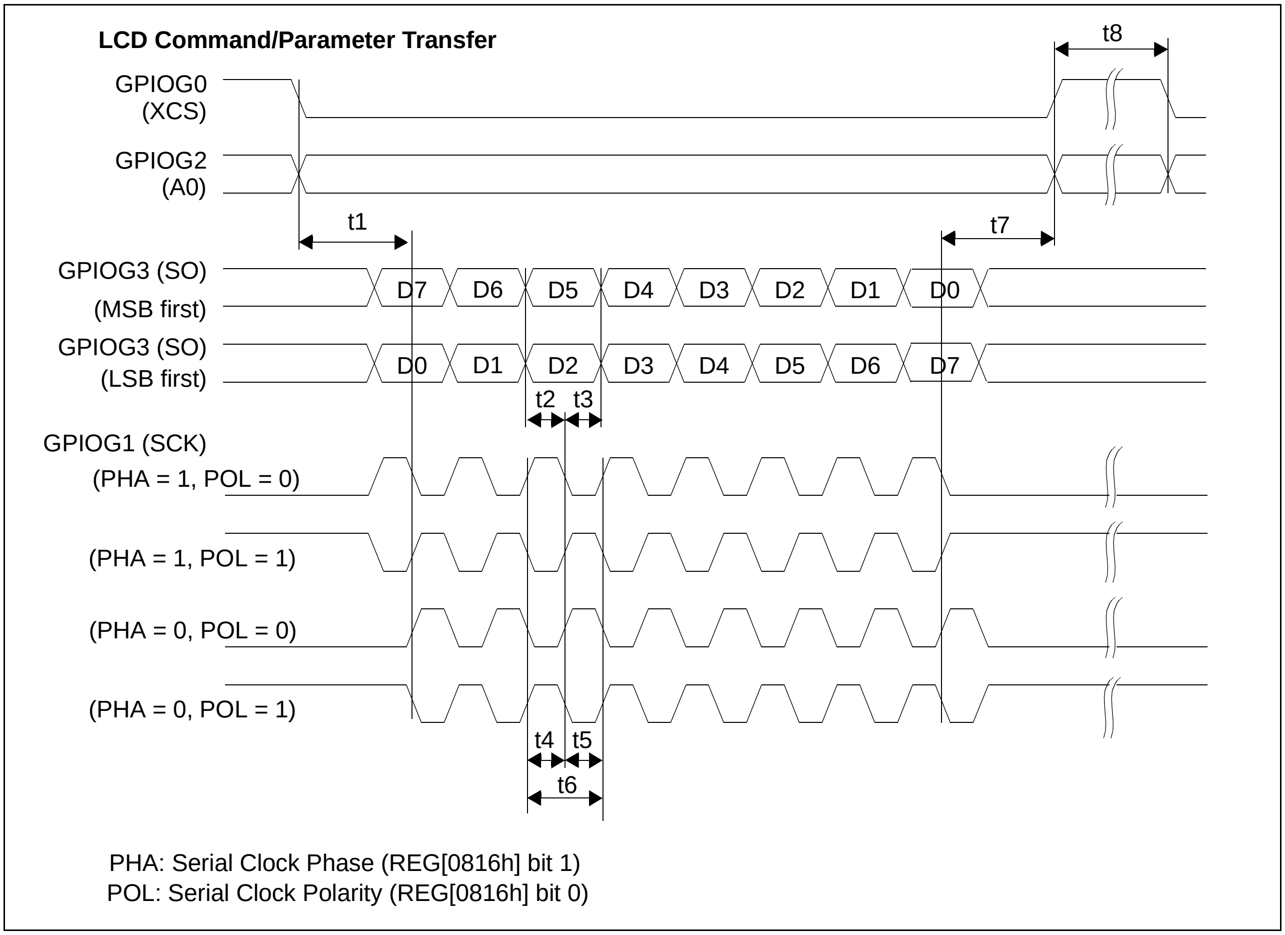


Figure 7-37: ND-TFD 8-Bit Serial Interface Timing

Table 7-38: ND-TFD 8-Bit Serial Interface Timing

Symbol	Parameter	Min	Typ	Max	Units
t1	Chip select setup time	1.5Ts - 1	1.5	1.5Ts + 1	Ts (Note 1)
t2	Data setup time	0.5Ts - 1	0.5	0.5Ts + 1	Ts
t3	Data hold time	0.5Ts - 1	0.5	0.5Ts + 1	Ts
t4	Serial clock pulse width low (high)	—	0.5	—	Ts
t5	Serial clock pulse width high (low)	—	0.5	—	Ts
t6	Serial clock period	—	1	—	Ts
t7	Chip select hold time for command/parameter transfer	1.5Ts - 1	1.5	1.5Ts + 1	Ts
t8	Chip select de-assert to reassert	—	1	—	Ts

1. Ts = Serial clock period

7.6.4 ND-TFD 9-Bit Serial Interface Timing

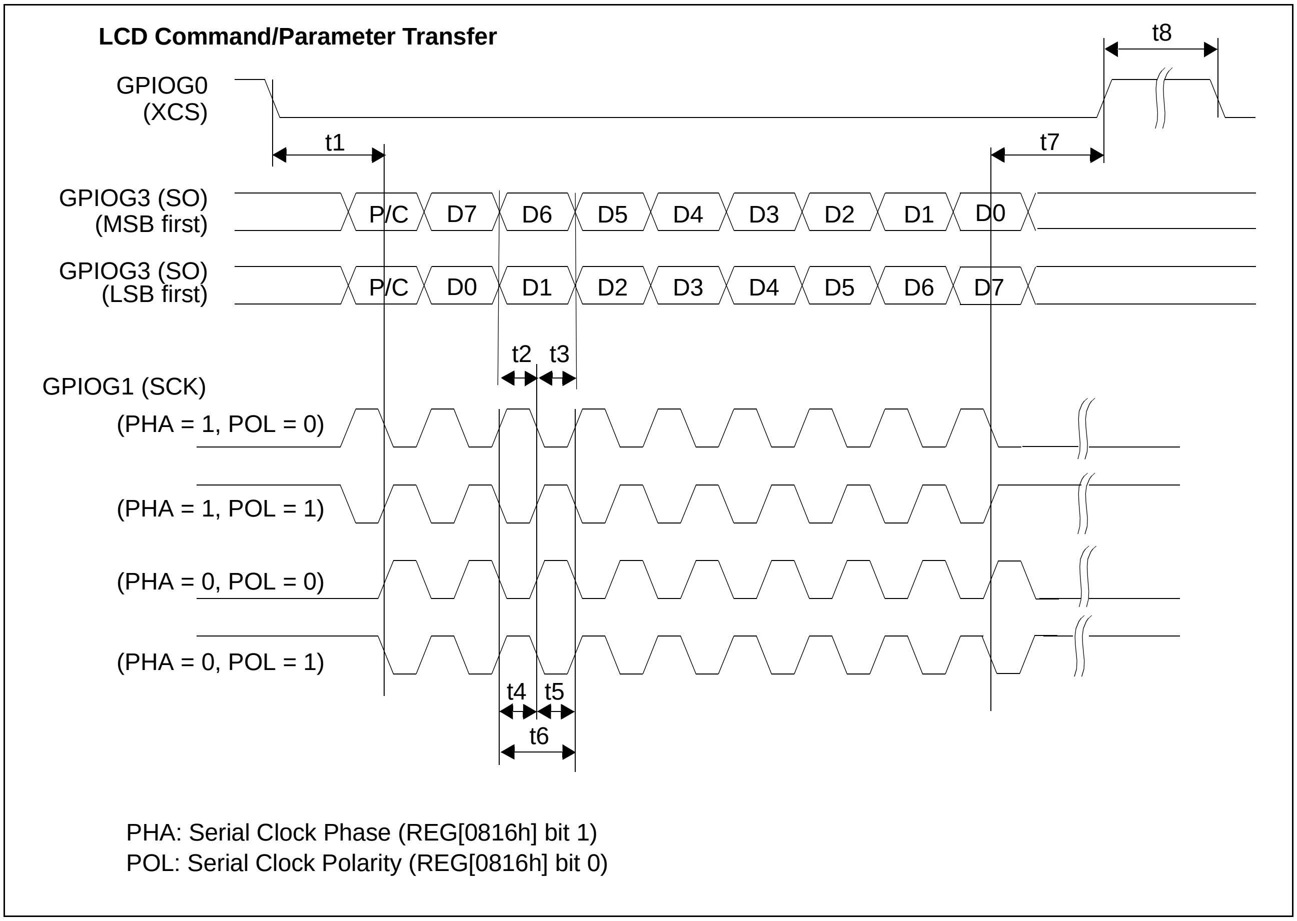


Figure 7-38: ND-TFD 9-Bit Serial Interface Timing

Table 7-39: ND-TFD 9-Bit Serial Interface Timing

Symbol	Parameter	Min	Typ	Max	Units
t1	Chip select setup time	1.5Ts - 1	1.5	1.5Ts + 1	Ts (Note 1)
t2	Data setup time	0.5Ts - 1	0.5	0.5Ts + 1	Ts
t3	Data hold time	0.5Ts - 1	0.5	0.5Ts + 1	Ts
t4	Serial clock pulse width low (high)	—	0.5	—	Ts
t5	Serial clock pulse width high (low)	—	0.5	—	Ts
t6	Serial clock period	—	1	—	Ts
t7	Chip select hold time for command/parameter transfer	1.5Ts - 1	1.5	1.5Ts + 1	Ts
t8	Chip select de-assert to reassert	—	1	—	Ts

1. Ts = Serial clock period

7.6.5 a-Si TFT Serial Interface Timing

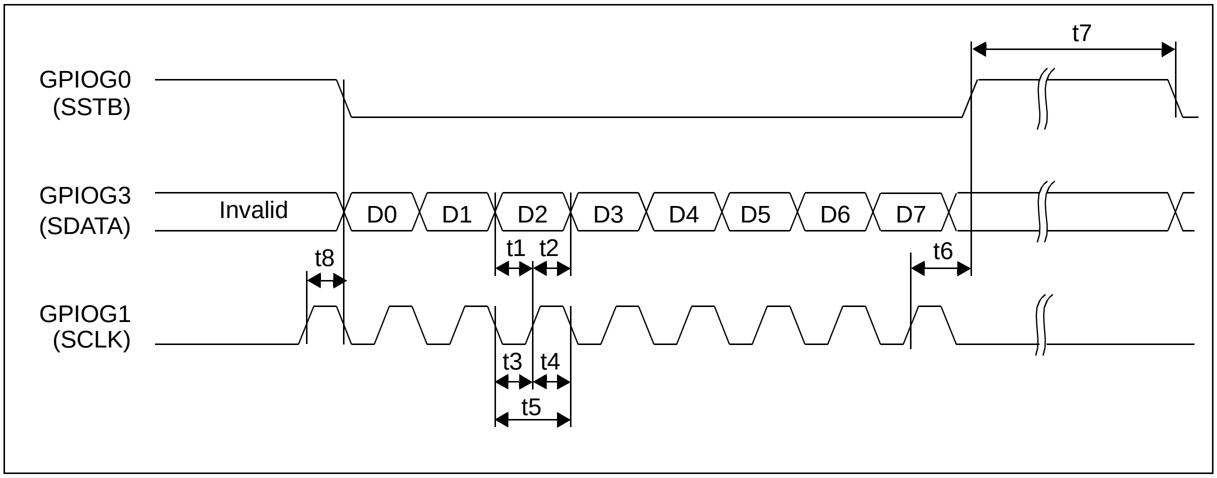


Figure 7-39: a-Si TFT Serial Interface Timing

Table 7-40: a-Si TFT Serial Interface Timing

Symbol	Parameter	Min	Typ	Max	Units
t1	Data Setup Time	0.5Ts - 1	0.5	0.5Ts + 1	Ts (Note 1)
t2	Data Hold Time	0.5Ts - 1	0.5	0.5Ts + 1	Ts
t3	Serial clock plus low period	—	0.5	—	Ts
t4	Serial clock pulse high period	—	0.5	—	Ts
t5	Serial clock period	—	1	—	Ts
t6	Chip select hold time	1.5Ts - 1	1.5	1.5Ts + 1	Ts
t7	Chip select de-assert to reassert	—	Note 2	—	Ts
t8	SCLK rising edge to SSTB falling edge	0.5Ts - 1	—	0.5Ts + 1	Ts

1. Ts = Serial clock period
2. This setting depends on software.

7.6.6 uWIRE Serial Interface Timing

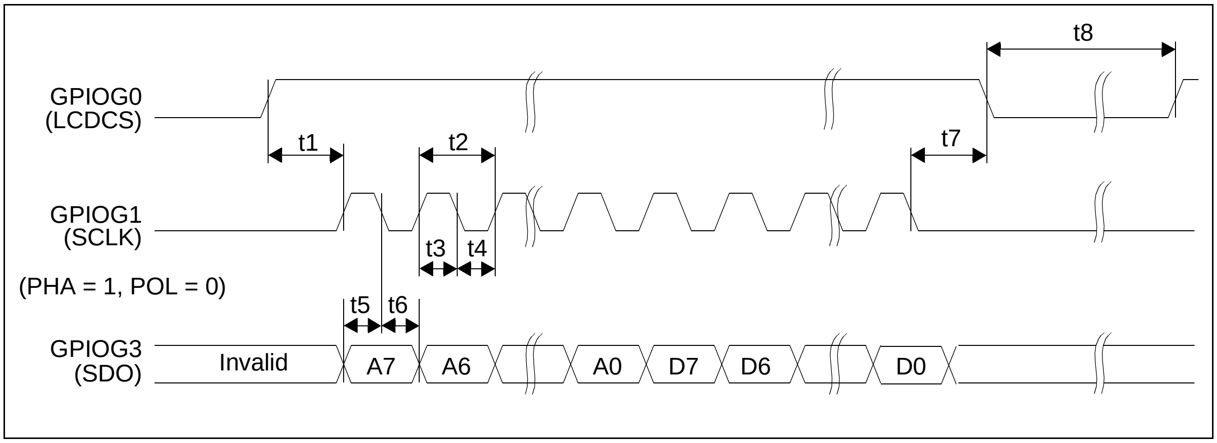


Figure 7-40: uWIRE Serial Interface Timing

Table 7-41: uWIRE Serial Interface Timing

Symbol	Parameter	Min	Typ	Max	Units
t1	Chip select setup time	1.0Ts - 1	1	1.0Ts + 1	Ts (Note 1)
t2	Serial clock Period	—	1	—	Ts
t3	Serial clock pulse width low	0.5Ts - 1	0.5	0.5Ts + 1	Ts
t4	Serial clock pulse width high	0.5Ts - 1	0.5	0.5Ts + 1	Ts
t5	Data setup time	0.5Ts - 1	0.5	0.5Ts + 1	Ts
t6	Data hold time	0.5Ts - 1	0.5	0.5Ts + 1	Ts
t7	Chip select hold time	1.5Ts - 1	1.5	1.5Ts + 1	Ts
t8	Chip select de-assert to reassert	—	Note 2	—	Ts

1. Ts = Serial clock period
2. This setting depends on software

7.6.7 24-bit Serial Interface Timing

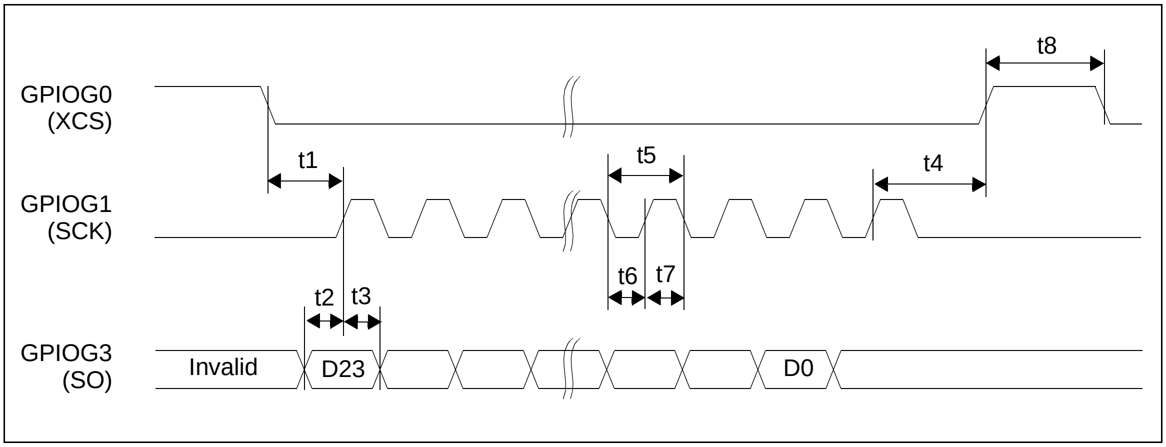


Figure 7-41: 24-bit Serial Interface Timing

Table 7-42: 24-bit Serial Interface Timing

Symbol	Parameter	Min	Typ	Max	Units
t1	Chip select setup time	1.5Ts - 1	—	1.5Ts + 1	ns (Note 1)
t2	Data setup time	0.5Ts - 1	—	0.5Ts + 1	ns
t3	Data hold time	0.5Ts - 1	—	0.5Ts + 1	ns
t4	Chip select hold time	1.5Ts - 1	—	1.5Ts + 1	ns
t5	Serial clock period	—	1	—	ns
t6	Serial clock pulse low	—	0.5	—	ns
t7	Serial clock pulse high	—	0.5	—	ns
t8	Chip select de-assert to re-assert	—	Note 2	—	ns

1. Ts = Serial clock period
2. This setting depends on software.

7.6.8 YUV Digital Output

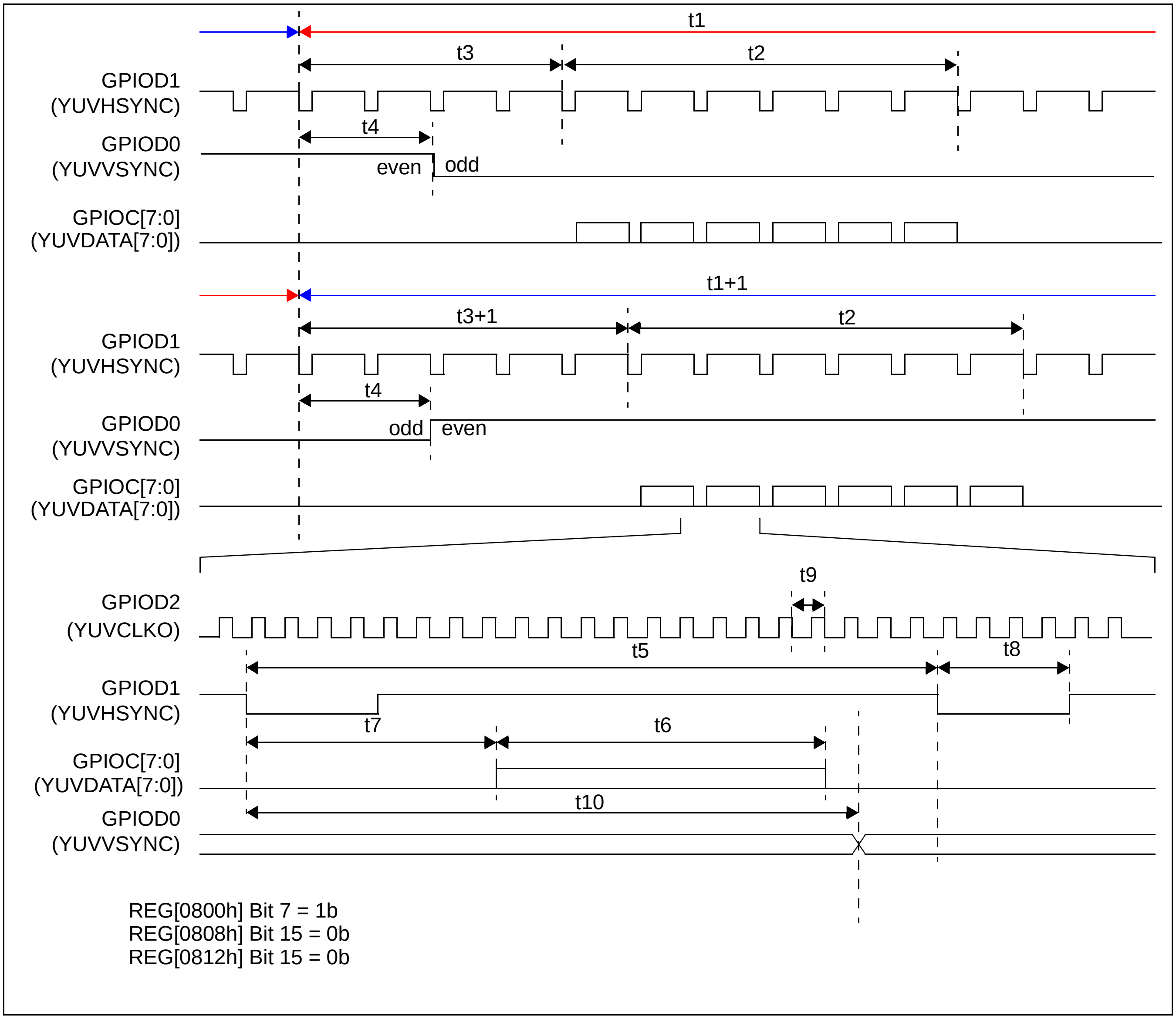


Figure 7-42: YUV Digital Output Interface Timing

Table 7-43: YUV Digital Output Interface Timing

Symbol	Parameter	Min	Typ	Max	Units
t1	Vertical Total (Even Field)	—	Note 2	—	Lines
t2	Vertical Display Period (Field)	—	Note 3	—	Lines
t3	Vertical Display Start Position (Field)	—	Note 4	—	Lines
t4	YUVVSYNC (GPIOD0) Signal Toggle Position	—	Note 5	—	Lines
t5	Horizontal Total	—	Note 6	—	CLOCK (Note 1)
t6	Horizontal Display Period	—	Note 7	—	CLOCK
t7	Horizontal Display Start Position	—	Note 8	—	CLOCK
t8	YUVHSYNC (GPIOD1) Horizontal Display Sync Pulse Width	—	Note 9	—	CLOCK
t9	YUVCLKO (GPIOD2) cycle	—	1	—	CLOCK
t10	YUVHSYNC (GPIOD1) Start Position	—	Note 10	—	CLOCK

1. CLOCK = YUVCLKO period
2. t1typ = REG[080Ch] bits 11-0 + 1
3. t2typ = REG[080Eh] bits 11-0 + 1
4. t3typ = REG[0810h] bits 11-0 + 1
5. t4typ = REG[0814h] bits 11-0
6. t5typ = REG[0802h] bits 11-0 + 1
7. t6typ = (REG[0804h] bits 10-0 + 1) x 8
8. t7typ = REG[0806h] bits 11-0 + 1
9. t8typ = REG[0808h] bits 8-0 + 1
10. t10typ = (REG[0802h] bits 11-0 + 1) - REG[080Ah] bits 11-0

Note

These are recommended settings when the ADV7170/ADV7177 video encoder is used with YUV Digital Output. System Clock should be set to 27MHz. When Panel Output is disabled, YUVCLKO, YUVHSYNC, YUVVSYNC, and YUVDATA[7:0] are driven low.

7.6.9 Single Monochrome 8-Bit Panel Timing

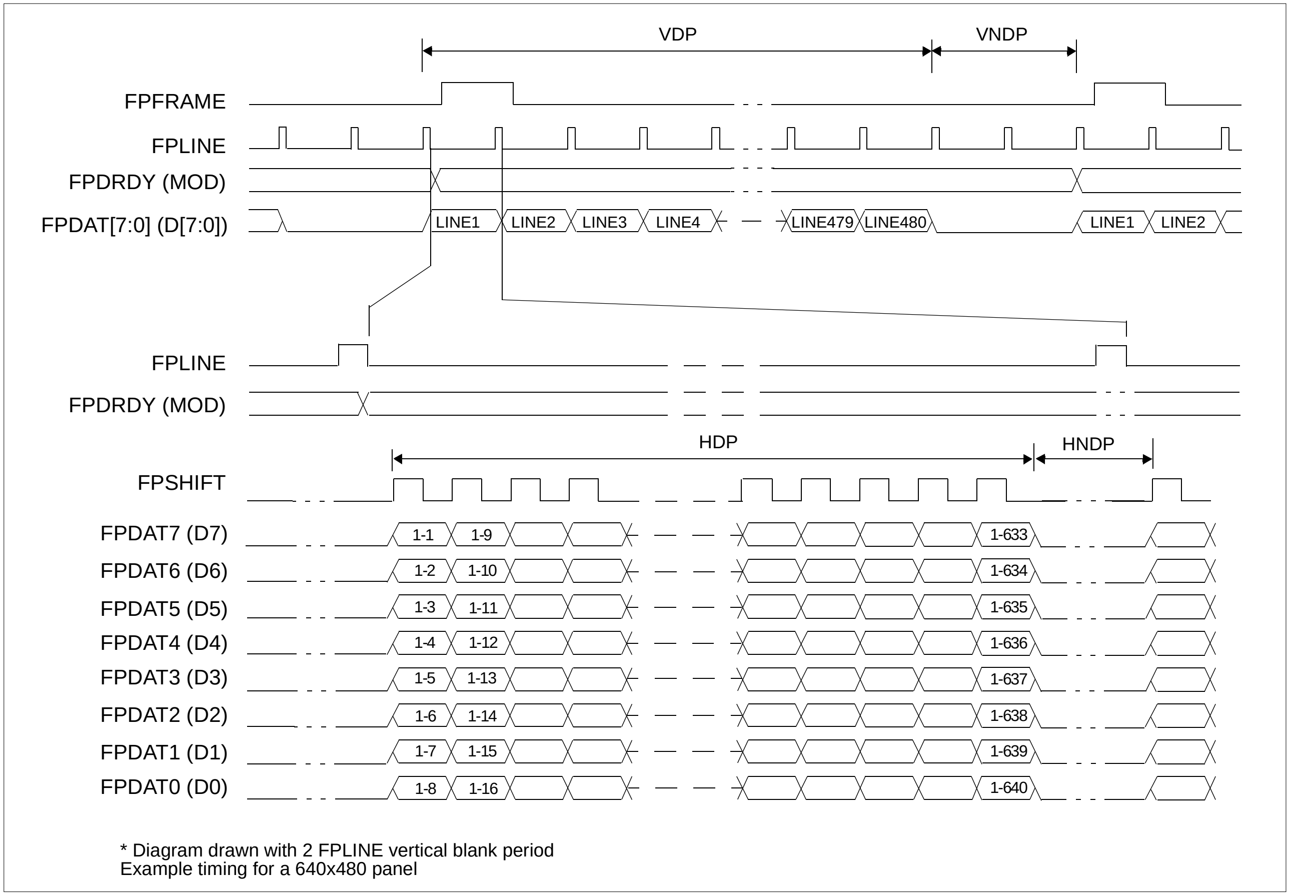


Figure 7-43: Single Monochrome 8-Bit Panel Timing

VDP = Vertical Display Period
= (REG[080Eh] bits 11-0) + 1 Lines
VNDP = Vertical Non-Display Period
= VT - VDP
= (REG[080Ch] bits 11-0) - (REG[080Eh] bits 11-0) Lines
HDP = Horizontal Display Period
= ((REG[0804h] bits 9-0) + 1) x 2 Tfpshift
HNDP = Horizontal Non-Display Period
= [((REG[0802h] bits 11-0) + 1) - (((REG[0804h] bits 9-0) + 1) x 2)] ÷ 8Tfpshift

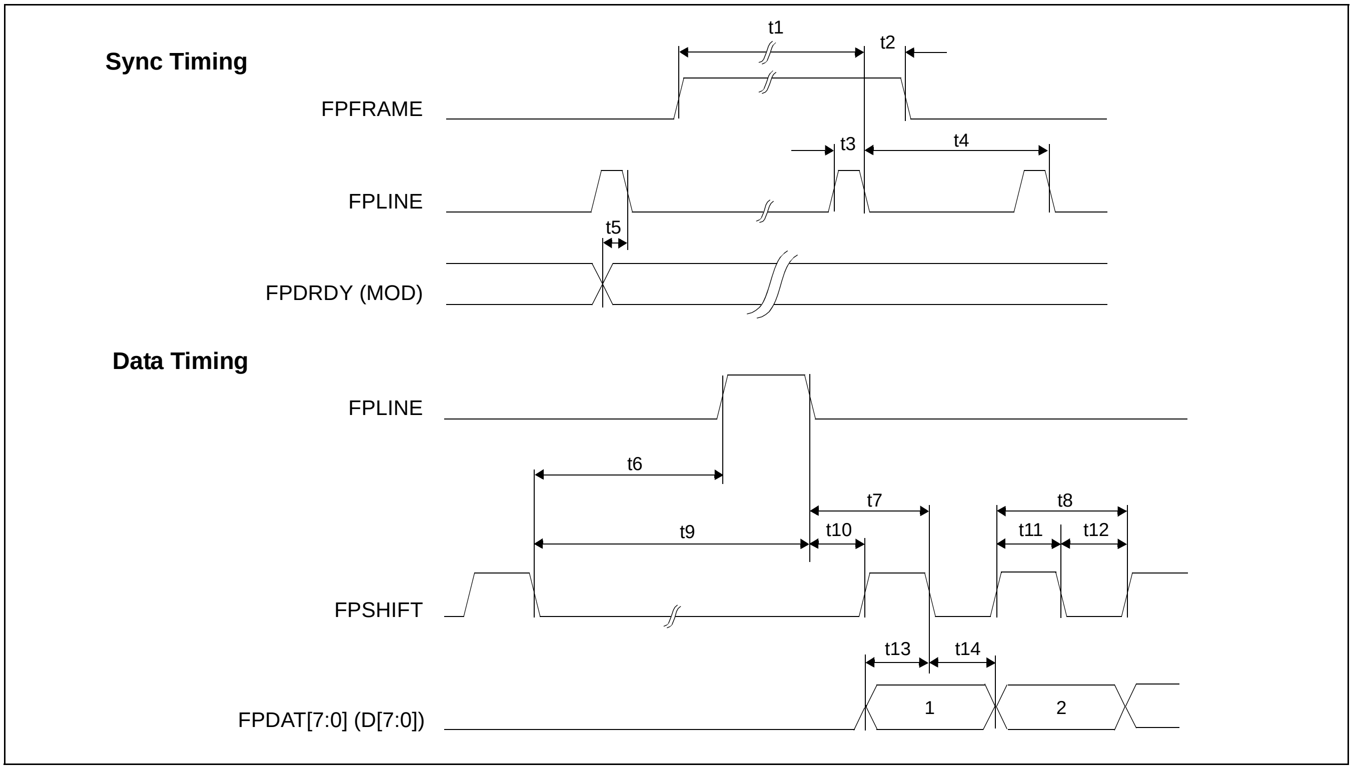


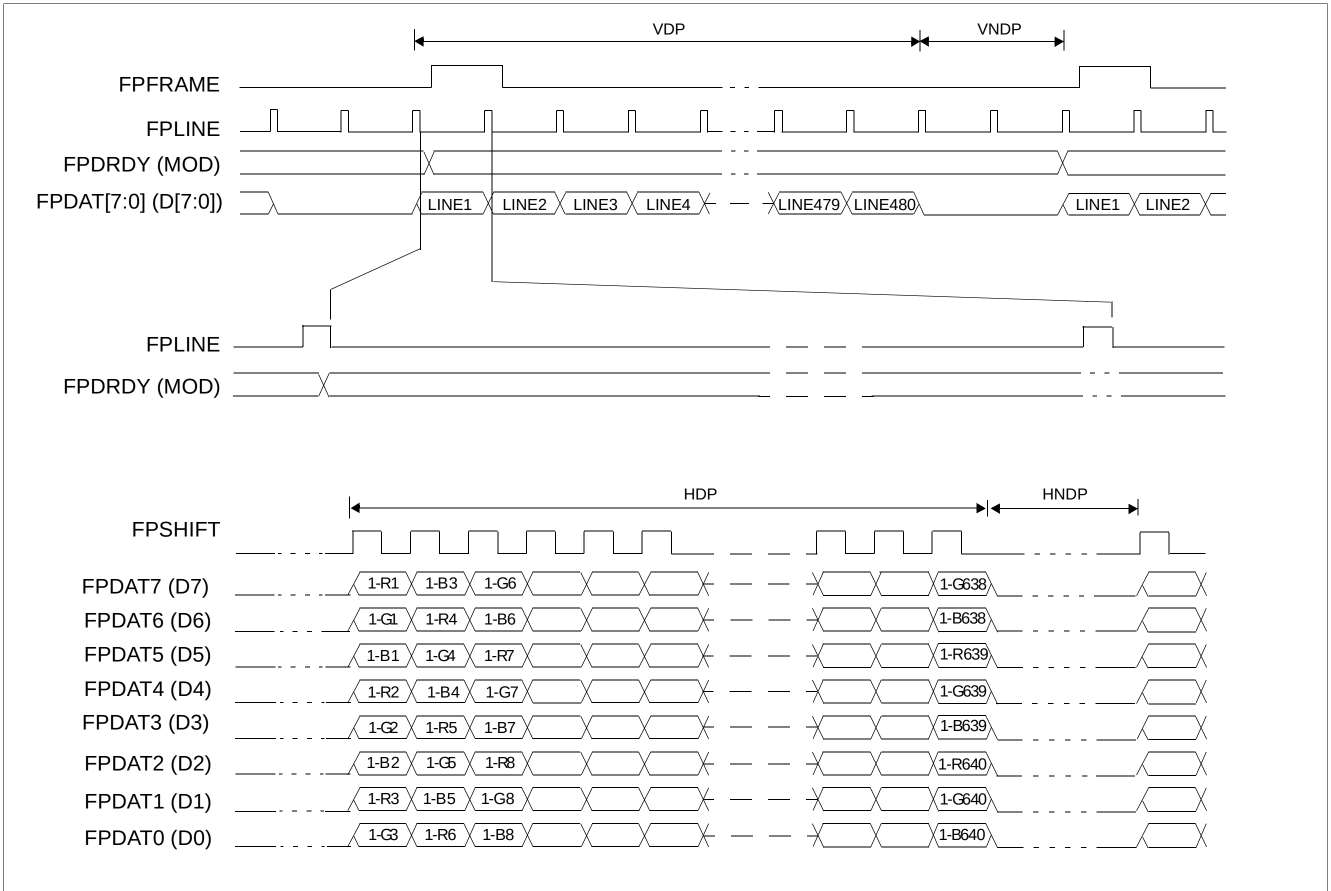
Figure 7-44: Single Monochrome 8-Bit Panel A.C. Timing

Table 7-44: Single Monochrome 8-Bit Panel A.C. Timing

Symbol	Parameter	Min	Typ	Max	Units
t1	FPFRAME setup to FPLINE falling edge		Note 3		Tfpshift
t2	FPFRAME hold from FPLINE falling edge		Note 4		Tfpshift
t3	FPLINE pulse width		Note 5		Tfpshift
t4	FPLINE period		Note 6		Tfpshift
t5	MOD delay from FPLINE falling edge		Note 7		Tfpshift
t6	FPSHIFT falling edge to FPLINE rising edge		Note 8		Tfpshift
t7	FPLINE falling edge to FPSHIFT falling edge		Note 9		Tfpshift
t8	FPSHIFT period		8		Ts
t9	FPSHIFT falling edge to FPLINE falling edge		Note 10		Tfpshift
t10	FPLINE falling edge to FPSHIFT rising edge		Note 11		Tfpshift
t11	FPSHIFT pulse width high		4		Ts
t12	FPSHIFT pulse width low		4		Ts
t13	FPDAT[7:0] (D[7:0]) setup to FPSHIFT falling edge		4		Ts
t14	FPDAT[7:0] (D[7:0]) hold to FPSHIFT falling edge		4		Ts

1. Ts = LCD pixel clock period
2. Tfpshift = Ts x 8
3. t1 = (VPW - HT) + HPS + HPW
   = [(REG[0812h] bits 4-0) x t4] + [(REG[080Ah] bits 11-0) ÷ 8] + t3
4. t2 = HT - HPS - HPW
   = t4 - [(REG[080Ah] bits 11-0) ÷ 8] - t3
5. t3 = HPW
   = ((REG[0808h] bits 8-0) + 1) ÷ 8
6. t4 = HT
   = ((REG[0802h] bits 11-0) + 1) ÷ 8
7. t5 = t3
8. t6 = HT - (HDP + HDPS) + 0.5
   = t4 - [((REG[0804h] bits 9-0) + 1) x 2 ÷ 8 - ((REG[0806h] bits 11-0) + 1) ÷ 8] + 0.5
9. t7 = t10 + 0.5
10. t9 = t6 + t3
11. t10 = HDPS - HPW
    = [((REG[0806h] bits 11-0) + 1) ÷ 8] - t3

7.6.10 Single Color 8-Bit Panel Timing (Format 2)



* Diagram drawn with 2 FPLINE vertical blank period
Example timing for a 640x480 panel

Figure 7-45: Single Color 8-Bit Panel Timing (Format 2)

VDP = Vertical Display Period
= (REG[080Eh] bits 11-0) + 1 Lines
VNDP = Vertical Non-Display Period
= VT - VDP
= (REG[080Ch] bits 11-0) - (REG[080Eh] bits 11-0) Lines
HDP = Horizontal Display Period
= ((REG[0804h] bits 9-0) + 1) x 2 x (3/8) FPSHIFTs
HNDP = Horizontal Non-Display Period
= [((REG[0802h] bits 11-0) + 1) - (((REG[804h] bits 9-0) + 1) x 3)] ÷ 4 FPSHIFTs

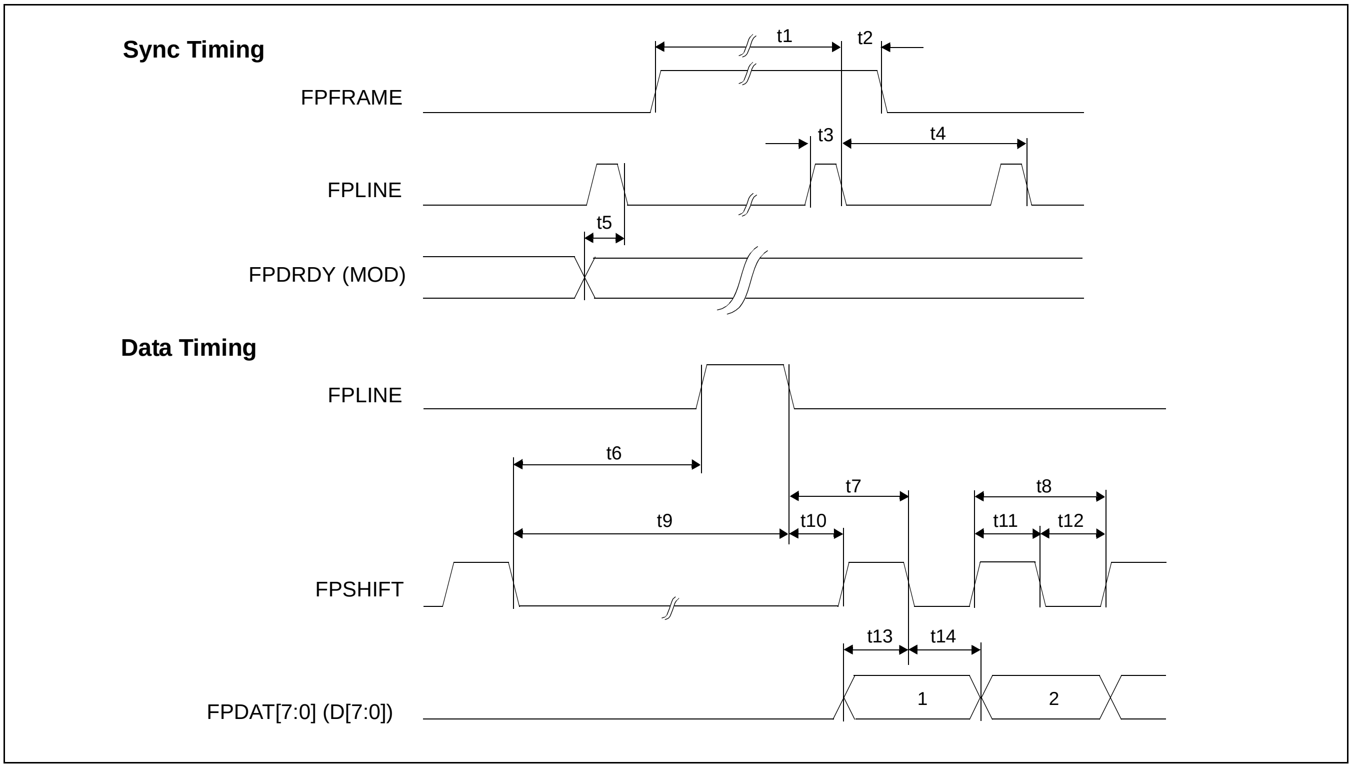


Figure 7-46: Single Color 8-Bit Panel A.C. Timing (Format 2)

Table 7-45: Single Color 8-Bit Panel A.C. Timing (Format 2)

Symbol	Parameter	Min	Typ	Max	Units
t1	FPFRAME setup to FPLINE falling edge		Note 3		Tfpshift
t2	FPFRAME hold from FPLINE falling edge		Note 4		Tfpshift
t3	FPLINE pulse width		Note 5		Tfpshift
t4	FPLINE period		Note 6		Tfpshift
t5	MOD delay to FPLINE falling edge		Note 7		Tfpshift
t6	FPSHIFT falling edge to FPLINE rising edge		Note 8		Tfpshift
t7	FPLINE falling edge to FPSHIFT falling edge		Note 9		Tfpshift
t8	FPSHIFT period		4		Ts
t9	FPSHIFT falling edge to FPLINE falling edge		Note 10		Tfpshift
t10	FPLINE falling edge to FPSHIFT rising edge		Note 11		Tfpshift
t11	FPSHIFT pulse width high		2		Ts
t12	FPSHIFT pulse width low		2		Ts
t13	FPDAT[7:0] (D[7:0]) setup to FPSHIFT falling edge		2		Ts
t14	FPDAT[7:0] (D[7:0]) hold to FPSHIFT falling edge	1	2		Ts

1. Ts = LCD pixel clock period
2. Tfpshift = Ts x 4
3. t1 = (VPW - HT) + HPS + HPW
   = [(REG[0812h] bits 4-0) x t4] + [(REG[080Ah] bits 11-0) ÷ 4] + t3
4. t2 = HT - HPS - HPW
   = t4 - [(REG[080Ah] bits 11-0) ÷ 4] - t3
5. t3 = HPW
   = ((REG[0808h] bits 8-0) + 1) ÷ 4
6. t4 = HT
   = ((REG[0802h] bits 11-0) + 1) ÷ 4
7. t5 = t3
8. t6 = HT - (HDP + HDPS) + 0.5
   = t4 - [((REG[0804h] bits 9-0) + 1) x 2 x (3 ÷ 8) - ((REG[0806h] bits 11-0) + 1) ÷ 4] + 0.5
9. t7 = t10 + 0.5
10. t9 = t6 + t3
11. t10 = HDPS - HPW
    = [((REG[0806h] bits 11-0) + 1) ÷ 4] - t3

7.7 Camera Interface Timing

7.7.1 Camera Interface YUV Timing (8-bit Data Bus Mode)

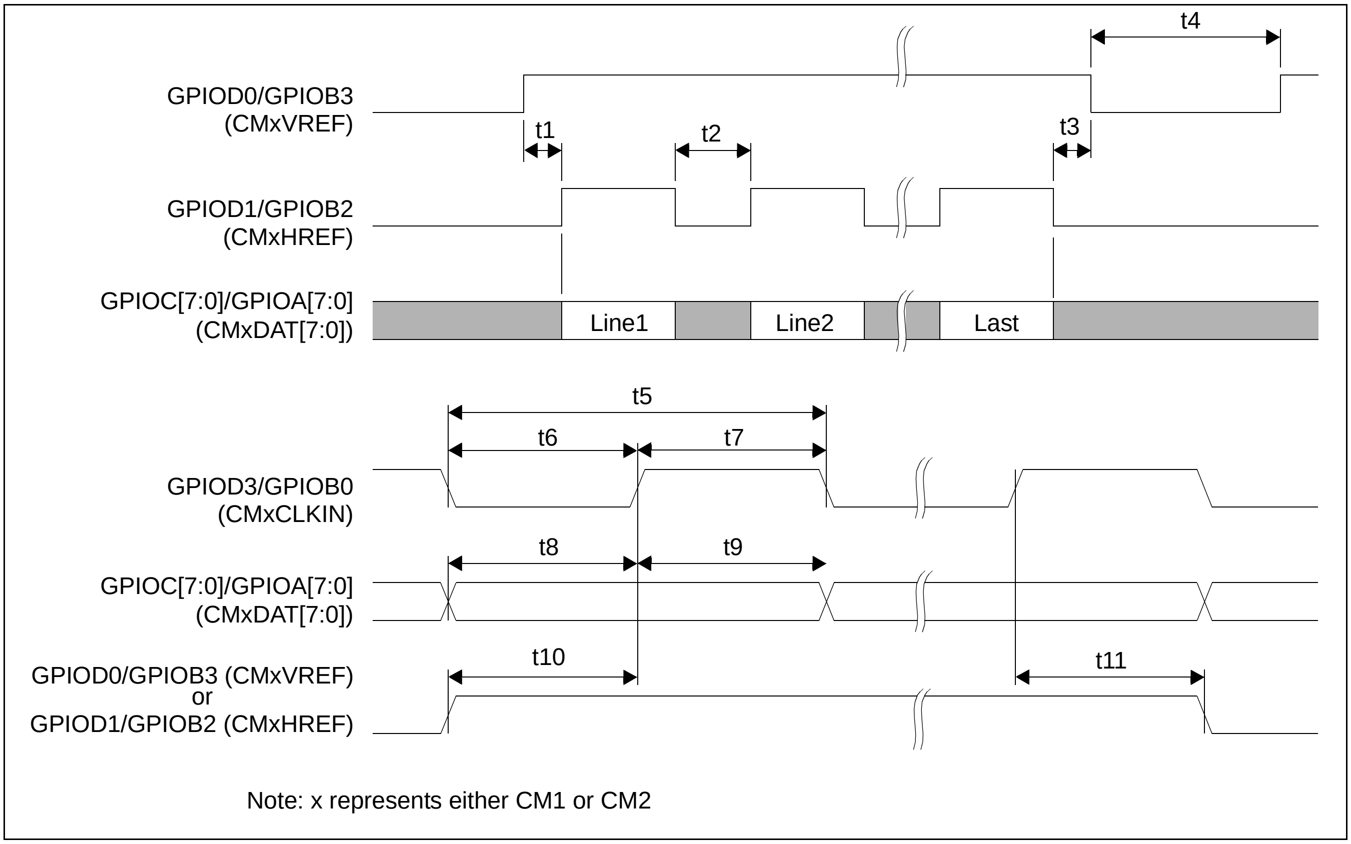


Figure 7-47: Camera Interface YUV Timing (8-bit data bus mode)

Table 7-46 : Camera Interface YUV Timing (8-bit data bus mode)

Symbol	Parameter	Min	Max	Units
t1	CMxVREF rising edge to CMxHREF rising edge	0	—	Tc (Note 1)
t2	Horizontal blank period	4	—	Tc
t3	CMxHREF falling edge to CMxVREF falling edge	0	—	Tc
t4	Vertical blank period	4	—	Line
t5	Camera input clock period	3	—	Ts (Note 2)
t6	Camera input clock pulse width low	1Ts+2	—	ns
t7	Camera input clock pulse width high	1Ts+2	—	ns
t8	Data setup time	10	—	ns
t9	Data hold time	10	—	ns
t10	CMxVREF, CMxHREF setup time	10	—	ns
t11	CMxVREF, CMxHREF hold time	10	—	ns

1. Tc = Camera block input clock period
2. Ts = System clock period

7.7.2 Camera Interface YUV Timing (16-bit data bus mode)

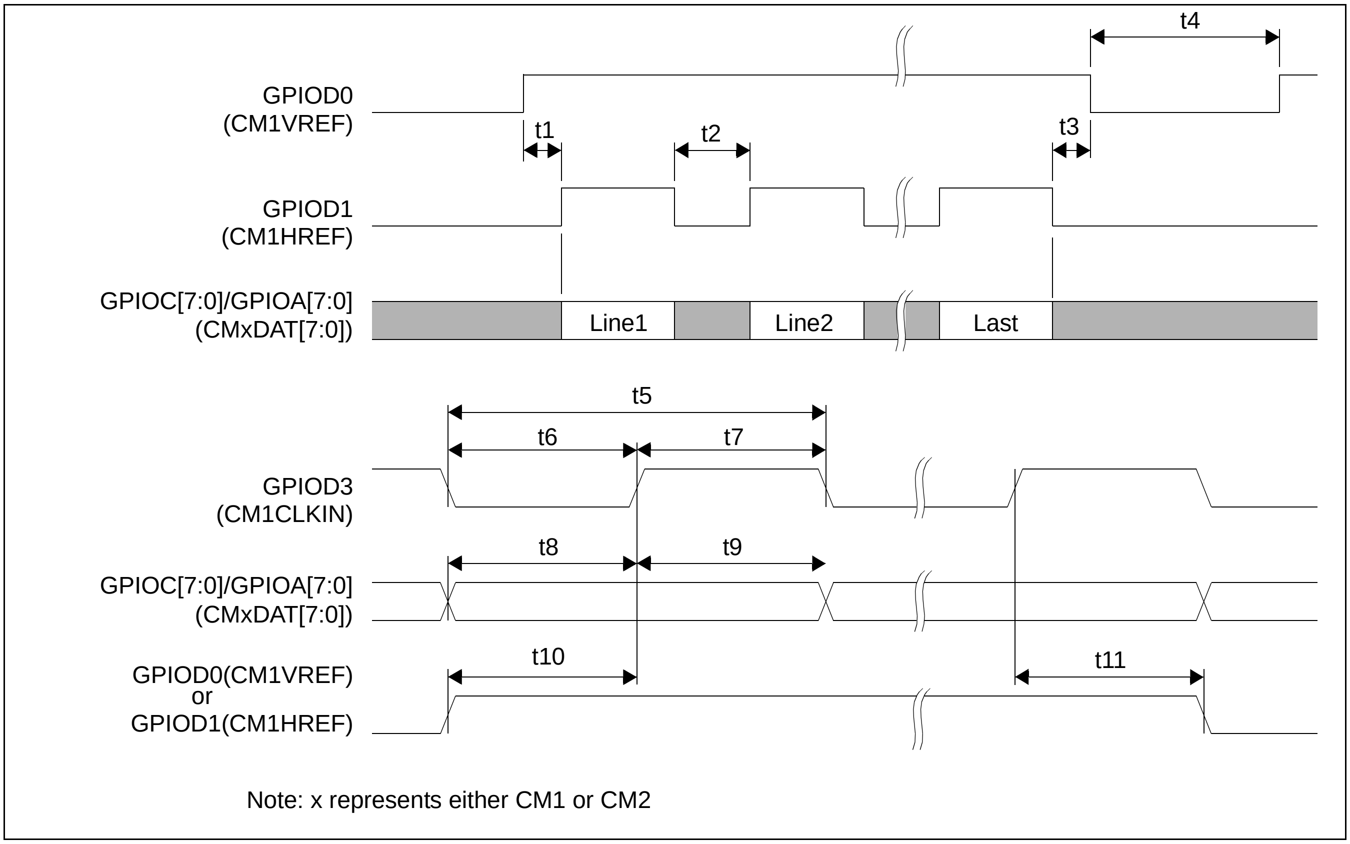


Figure 7-48: Camera Interface YUV Timing (16-bit data bus mode)

Table 7-47: Camera Interface YUV Timing (16-bit data bus mode)

Symbol	Parameter	Min	Max	Units
t1	CM1VREF rising edge to CM1HREF rising edge	0	—	Tc (Note 1)
t2	Horizontal blank period	4	—	Tc
t3	CM1HREF falling edge to CM1VREF falling edge	0	—	Tc
t4	Vertical blank period	4	—	Line
t5	Camera input clock period	6	—	Ts (Note 2)
t6	Camera input clock pulse width low	1Ts+2	—	ns
t7	Camera input clock pulse width high	1Ts+2	—	ns
t8	Data setup time	10	—	ns
t9	Data hold time	10	—	ns
t10	CM1VREF, CM1HREF setup time	10	—	ns
t11	CM1VREF, CM1HREF hold time	10	—	ns

1. Tc = Camera block input clock period
2. Ts = System clock period

7.7.3 Camera Interface JPEG Timing

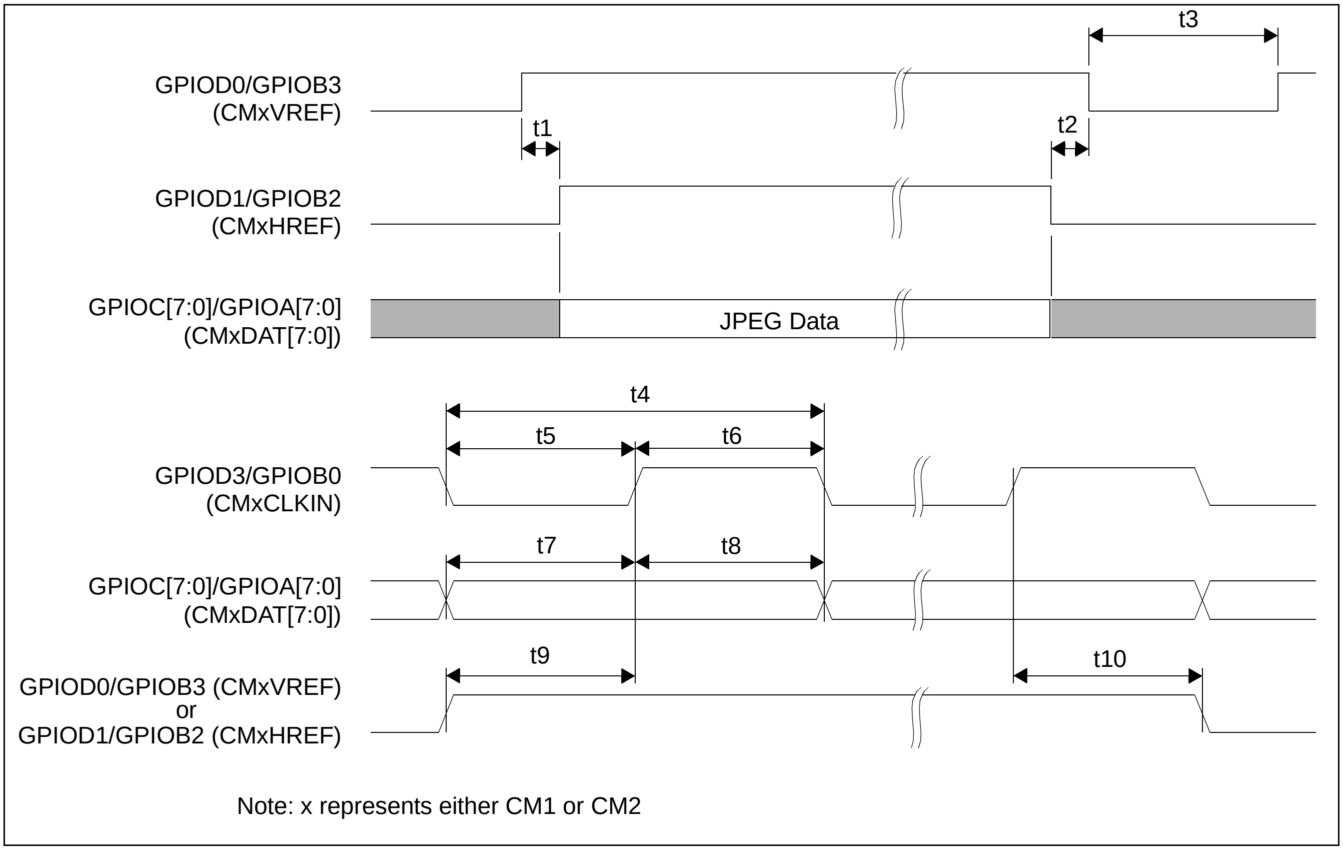


Figure 7-49 Camera Interface JPEG Timing

Table 7-48 Camera Interface JPEG Timing

Symbol	Parameter	Min	Typ	Max	Units
t1	CMxVREF rising edge to CMxHREF rising edge	0	—	—	Tc (Note 1)
t2	CMxHREF falling edge to CMxVREF falling edge	0	—	—	Tc
t3	Vertical blank period	1	—	—	Tc
t4	Camera input clock period Tc	3	—	—	Ts (Note 2)
t5	Camera input clock pulse width low	1.5	—	—	Ts
t6	Camera input clock pulse width high	1.5	—	—	Ts
t7	Data setup time	15	—	—	ns
t8	Data hold time	15	—	—	ns
t9	CMxVREF (CMxHREF) setup time	10	—	—	ns
t10	CMxVREF (CMxHREF) hold time	10	—	—	ns

1. Tc is the Camera block input clock period.
2. Ts is the System clock period.

7.7.4 Strobe Control Output Timing

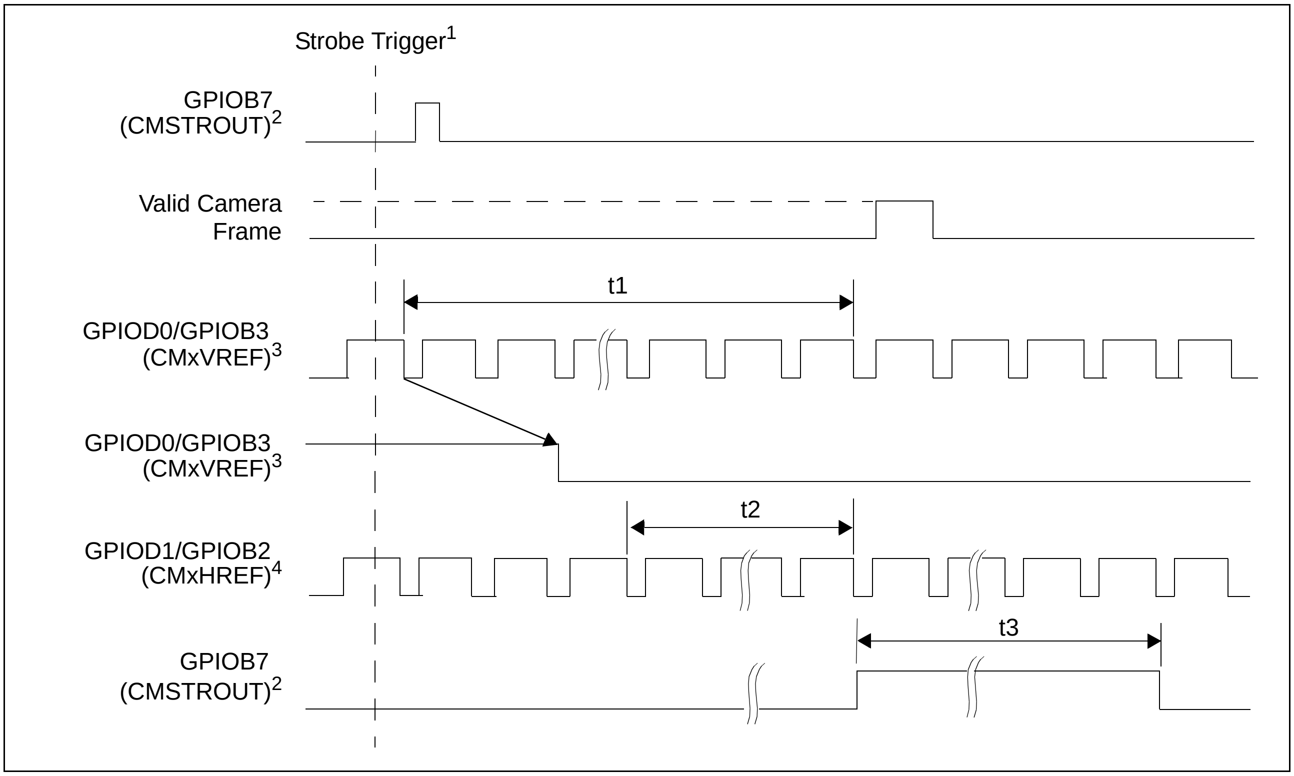


Figure 7-50: Strobe Control Output Timing

Note

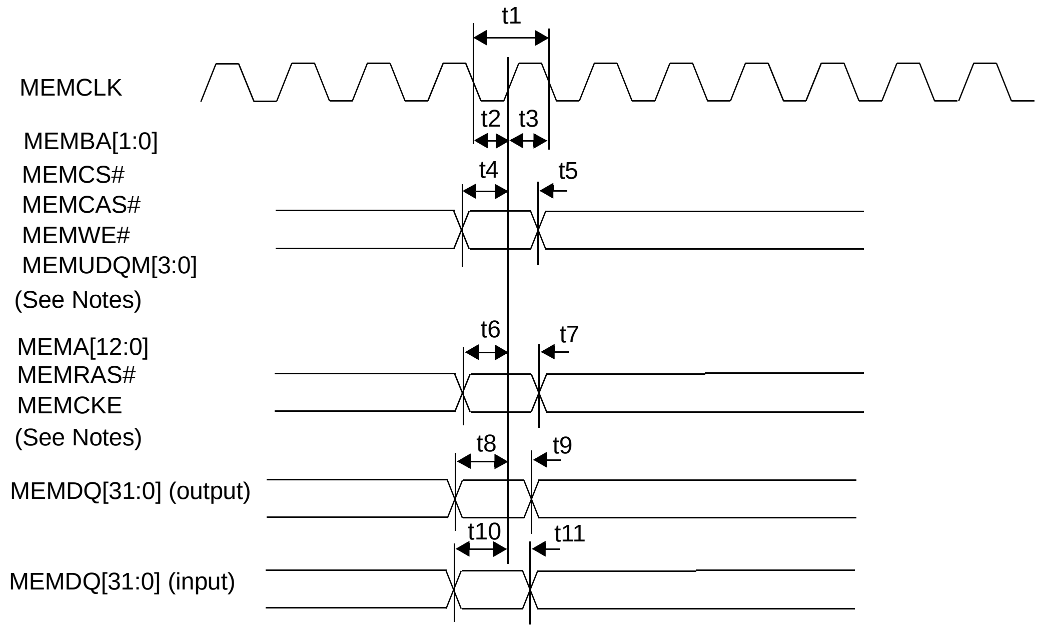
1. For more information on the strobe trigger, see the bit description for REG[2024h] bits 7-4 and Section 22.3, “Strobe Control Signal” on page 469.
2. CMSTROUT Active Select: High (REG[2024h] bits 3-0 = 1011b)
3. CMxVREF Active Select: Low (REG[2002h] bit 1 = 0b or REG[2006h] bit 1 = 0b)
4. CMxHREF Active Select: Low (REG[2002h] bit 2 = 0b or REG[2006h] bit 2 = 0b)

Table 7-49: Strobe Control Output Timing

Symbol	Parameter	Min	Typ	Max	Units
t1	CMxVREF delay from first CMxVREF falling edge (rising edge if active high) after the strobe trigger	—	Note 1	—	Tcmv (Note 4)
t2	CMxHREF delay from first CMxHREF falling edge (rising edge if active high) after CMxVREF active	—	Note 2	—	Tcmh (Note 5
t3	CMSTROUT active pulse width	—	Note 3	—	Tcmh

1. t1typ = REG[2024h] bits 7-4 (t1 is always 0 for single frame capture mode (REG[2012h] bit 6 = 1b) and REG[2024h] bits 7-4 are ignored)
2. t2typ = (REG[2020h] bits 15-0)
3. t3typ = (REG[2022h] bits 15-0) + 1
4. Tcmv = CMxVREF period
5. Tcmh = CMxHREF period

7.8 SDRAM Interface Timing



Notes:
Control signal1 outputs are MEMBA[1:0], MEMCS#, MEMCAS#, MEMWE#, and MEMUDQM[3:0].
Control signal2 outputs are MEMA[12:0], MEMRAS#, and MEMCKE.

Figure 7-51: SDRAM Setup/Hold to MEMCLK Timing

Table 7-50: SDRAM Setup/Hold to MEMCLK Timing

Symbol	Parameter	Min	Max	Units
t1	MEMCLK cycle time	8	—	ns
t2	MEMCLK low pulse width	3	—	ns
t3	MEMCLK high pulse width	3	—	ns
t4	Control signal1 outputs setup time to MEMCLK (see Note)	0.5Tc-1ns	—	Tc
t5	Control signal1 outputs hold time to MEMCLK (see Note)	0.5	—	Tc
t6	Control signal2 outputs setup time to MEMCLK (see Note)	0.5Tc-2ns	—	Tc
t7	Control signal2 outputs hold time to MEMCLK (see Note)	0.5	—	Tc
t8	MEMDQ[31:0] output setup time to MEMCLK	0.5Tc-2ns	—	Tc
t9	MEMDQ[31:0] output hold time to MEMCLK	0.5Tc-2ns	—	Tc
t10	MEMDQ[31:0] input setup time to MEMCLK (See Note)	3.6	—	ns
t11	MEMDQ[31:0] input hold time to MEMCLK	0	—	ns

1. Tc is a unit of MEMCLK cycle.
2. The MEMDQ[31:0] setup time (t10) is the value specified by REG[1C00h] = 11h.

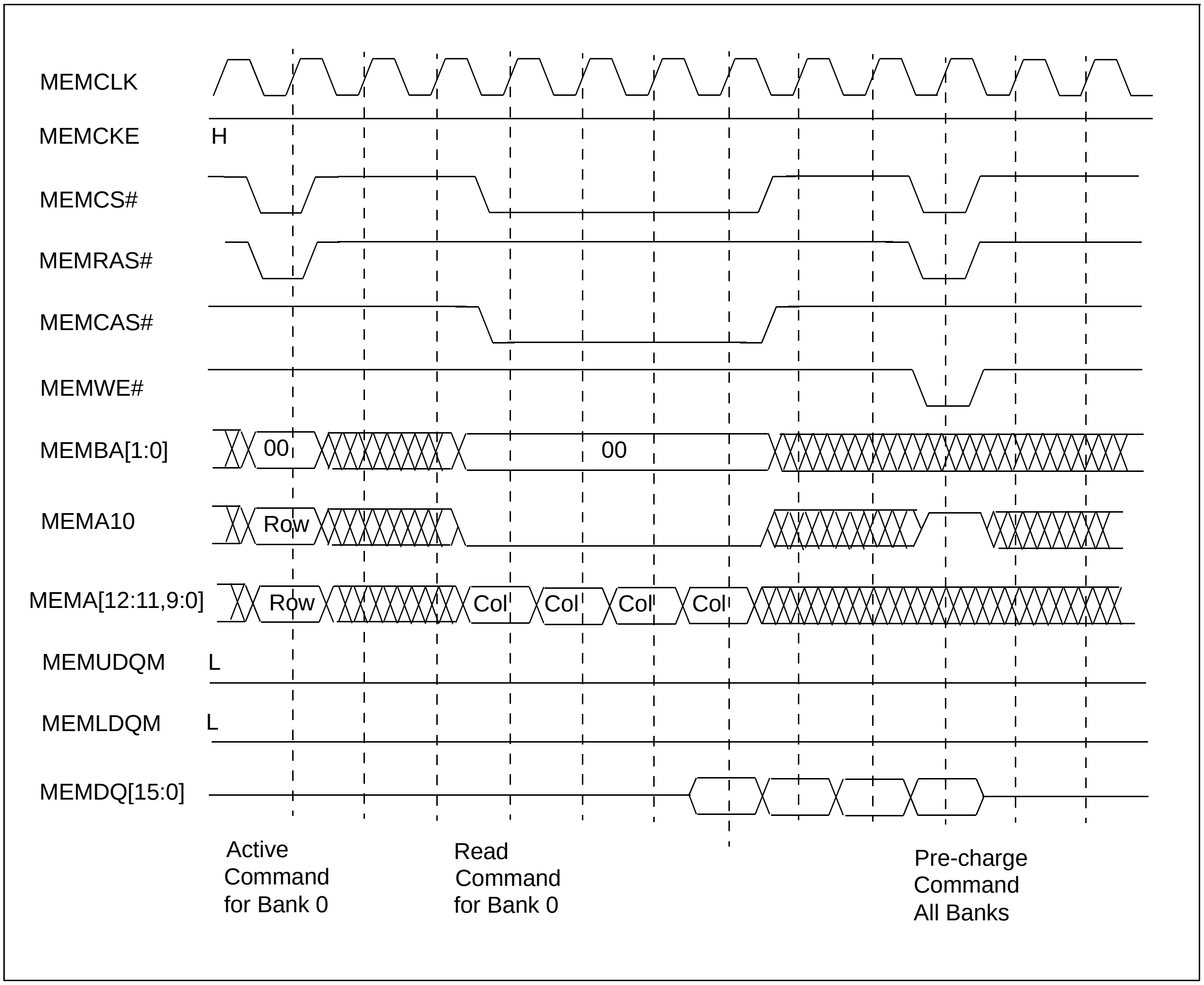


Figure 7-52: SDRAM Read Timing (Ex. Read Length = 4, CAS Latency = 3)

Note

Burst length is automatically set (undefined length).

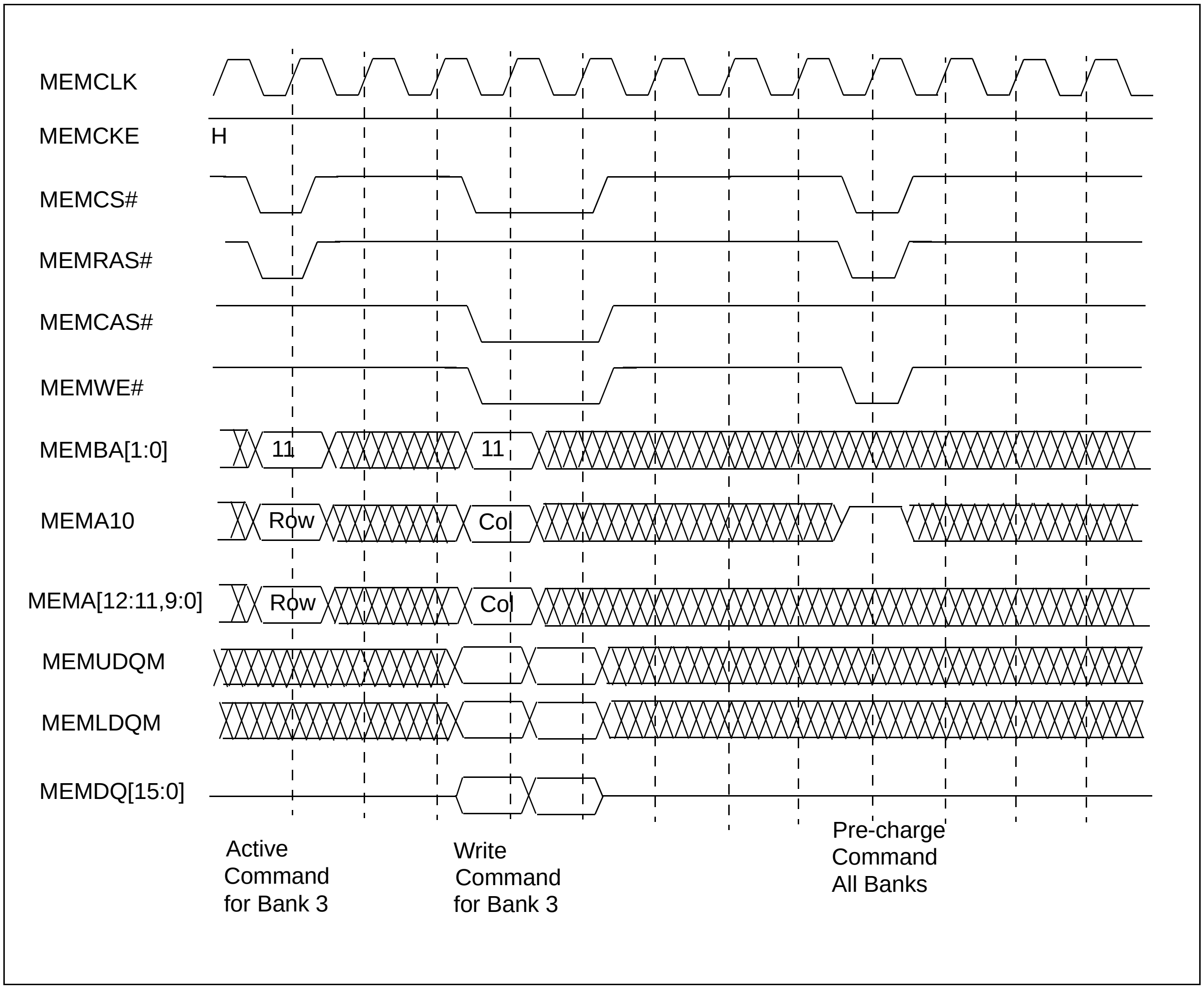


Figure 7-53: SDRAM Write Timing (Ex. Write Length = 2)

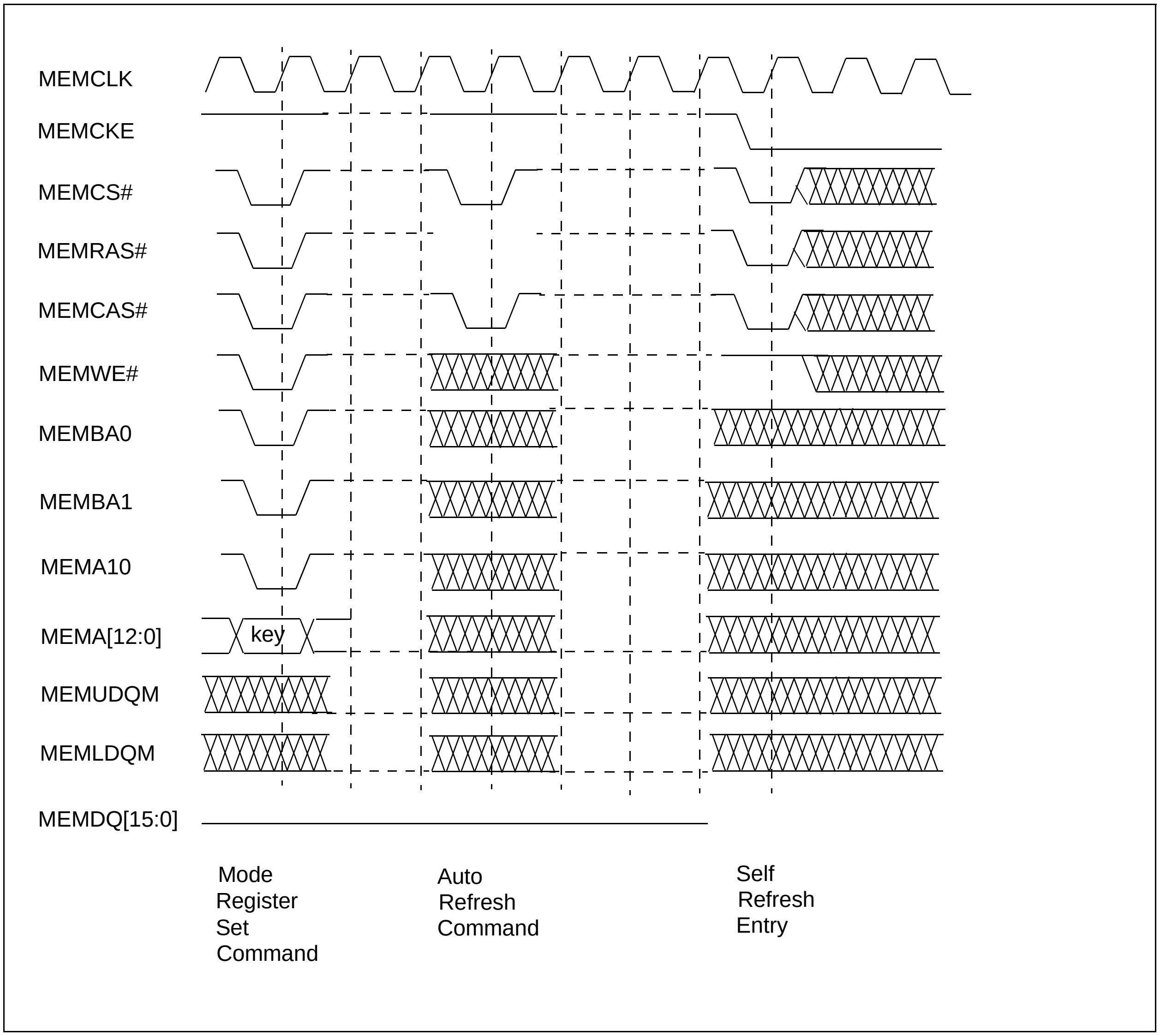


Figure 7-54: Mode Register Set / Auto Refresh / Auto Pre-charge Timing

7.9 I2C Interface Timing

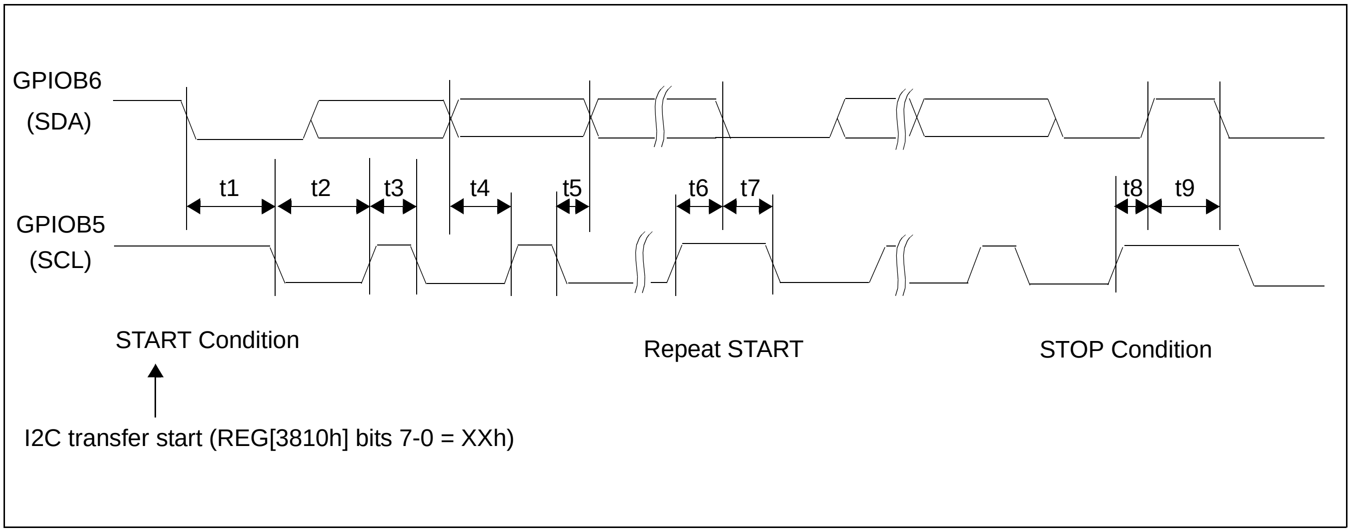


Figure 7-55: I2C Interface Timing

Note

Rise time of I2C SDA and I2C SCL is determined by the external pull-up resistor used and the load capacitance on the PCB.

Table 7-51: I2C Interface Timing for Standard Mode

Symbol	Parameter	Min	Typ	Max	Units
t1	Hold time START condition	—	Note 2	—	I2CCLK (Note 1)
t2	I2C SCL low time	—	Note 3	—	I2CCLK
t3	I2C SCL high time	—	Note 4	—	I2CCLK
t4	Read data setup before I2CDATACLK high	1	—	—	I2CCLK
t5	Data hold after I2C SCL low	1	—	—	I2CCLK
t6	Set-up time Repeated START condition	—	Note 5	—	I2CCLK
t7	Hold time Repeated START condition	Note 6	—	—	I2CCLK
t8	Set-up time for STOP condition	—	Note 7	—	I2CCLK
t9	Bus free time between a STOP and START condition	—	Note 8	—	I2CCLK

1. I2CCLK= I2C clock frequency = SYSCLK ÷ (REG[0430h] bits 7-0 + 1)
2. t1typ = (REG[3814h] bits 15-0) + 3
3. t2typ = (REG[3818h] bits 15-0) + 1
4. t3typ = (REG[3814h] bits 15-0) + 8
5. t6typ = (REG[3814h] bits 15-0) + (REG[3818h] bits 15-0) + 17
6. t7min = (REG[3814h] bits 15-0) + 2
7. t8typ = (REG[3814h] bits 15-0) + 7
8. t9typ = (REG[3818h] bits 15-0) + 10

Table 7-52: I2C Interface Timing for Fast Mode

Symbol	Parameter	Min	Typ	Max	Units
t1	Hold time START condition	—	Note 2	—	I2CCLK (Note 1)
t2	I2C SCL low time	—	Note 3	—	I2CCLK
t3	I2C SCL high time	—	Note 4	—	I2CCLK
t4	Read data setup before I2CDATACLK high	1	—	—	I2CCLK
t5	Data hold after I2C SCL low	1	—	—	I2CCLK
t6	Set-up time Repeated START condition	—	Note 5	—	I2CCLK
t7	Hold time Repeated START condition	Note 6	—	—	I2CCLK
t8	Set-up time for STOP condition	—	Note 7	—	I2CCLK
t9	Bus free time between a STOP and START condition	—	Note 8	—	I2CCLK

1. I2CCLK = I2C clock frequency = SYSCLK ÷ (REG[0430h] bits 7-0 + 1)
2. t1typ = (REG[381Ch] bits 15-0) + 3
3. t2typ = (REG[3820h] bits 15-0) + 1
4. t3typ = (REG[381Ch] bits 15-0) + 8
5. t6typ = (REG[381Ch] bits 15-0) + 135
6. t7min = (REG[381Ch] bits 15-0) + 2
7. t8typ = (REG[381Ch] bits 15-0) + 7
8. t9typ = 128

Note

The following registers have minimum values. If a value less than the minimum value is written to the register, the minimum value is automatically written to the register.

REG[3814h] min = 0006h
REG[3818h] min = 0008h
REG[381Ch] min = 0006h
REG[3820h] min = 0008h

7.10 Keypad Interface Timing

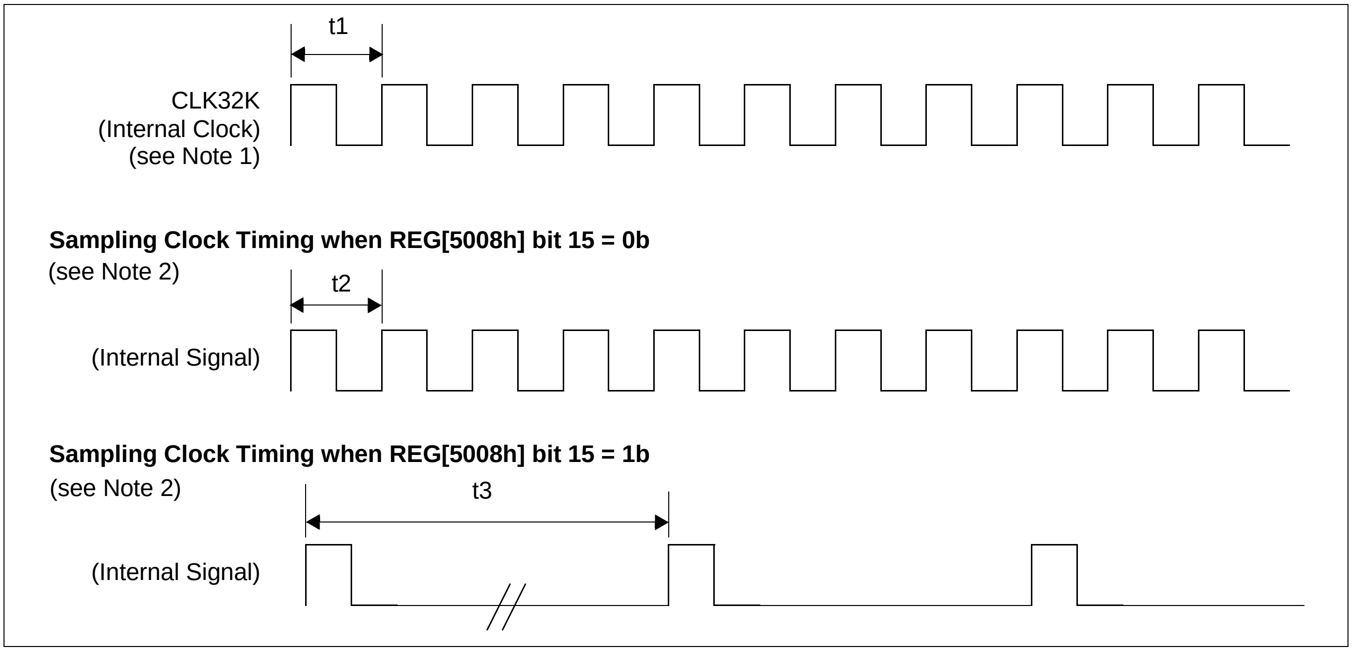

t1
CLK32K
(Internal Clock)
(see Note 1)
Sampling Clock Timing when REG[5008h] bit 15 = 0b
(see Note 2)
t2
(Internal Signal)
Sampling Clock Timing when REG[5008h] bit 15 = 1b
(see Note 2)
t3
(Internal Signal)

Figure 7-56: Keypad Interface Base Timing

Note

1. CLK32K is an internal case clock used for the Keypad interface. Users cannot see this clock.
2. Sampling Clock is the internal input sampling clock for the Keypad interface. Users cannot see this clock.

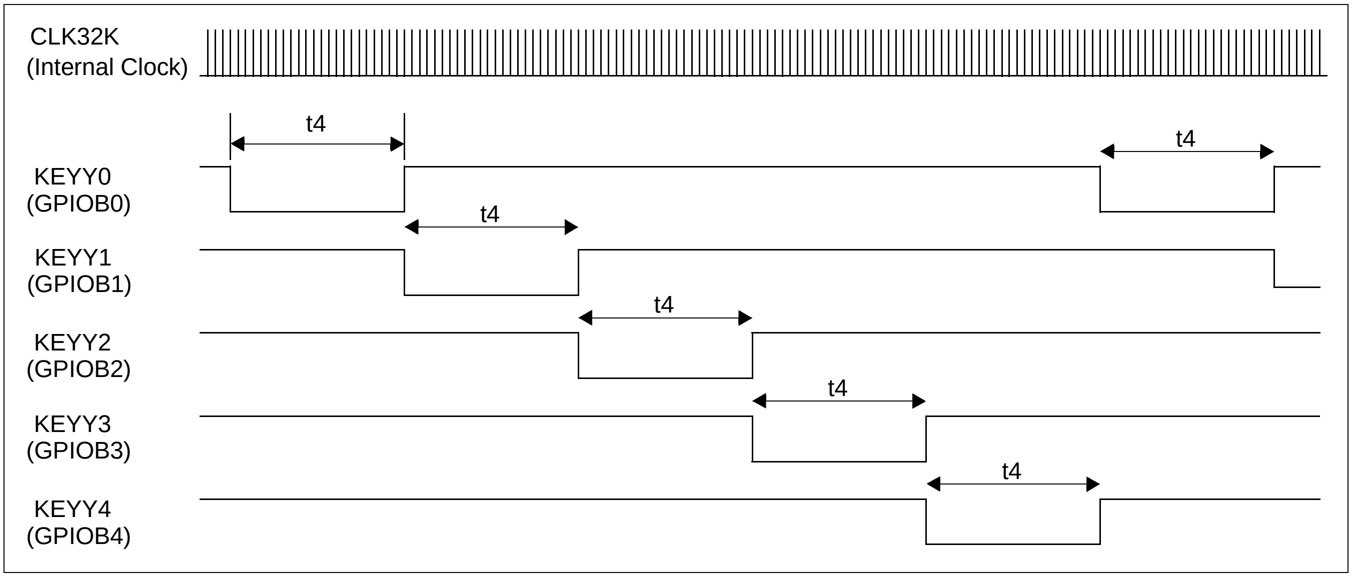


Figure 7-57: Keypad Interface Drive Timing

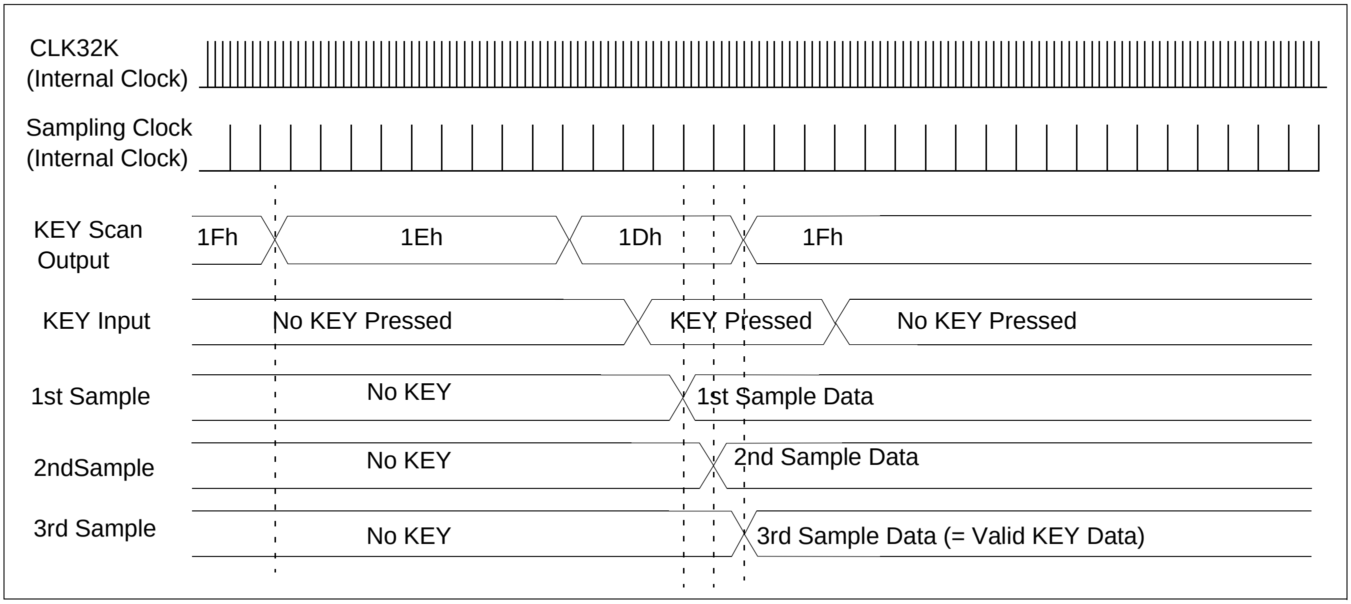


Figure 7-58: Keypad Interface Input Timing

Note

Keypad data is internally sampled three times. If all three samples are the same, the data is recognized as valid keypad input data.

Table 7-53 : Keypad Interface Timing

Symbol	Parameter	Min	Typ	Max	Units
f_{CLK32K}	32KHz equivalent clock frequency		32		KHz
t1	CLK32K period	—	$1/f_{CLK32K}$	—	us
t2	Sampling Clock pulse width (same as t1)	—	$1/f_{CLK32K}$	—	us
t3	Sampling Clock pulse width	—	Note 2	—	CLK32K
t4	Key Driving Period	—	Note 3	—	CLK32K

1) The frequency of f_{CLK32K} is specified by REG[042Ch].
2) t2typ, t3typ is specified by REG[5006h].
3) t4typ is specified by REG[500Ah].

8 Memory Map

The S1D13513 includes an SDRAM interface which supports either 64Mbit (8M bytes), 128Mbit (16M bytes), 256Mbit (32M bytes), or 512Mbit (64M bytes) of external SDRAM or external mobile SDRAM.

This memory must be accessed differently depending on the addressing method used.

Note

The S1D13513 host interface includes a memory buffer which accelerates memory accesses by “reading ahead”. For further details on this Read Ahead feature, see Section 20.4, “Read Ahead Feature” on page 442.

8.1 Accessing Memory using Direct Addressing

When direct addressing is selected (see Section 5.3, “Summary of Configuration Options” on page 41), the first Megabyte of external SDRAM can be addressed linearly. The remaining memory (up to 63M bytes) must be accessed using four 256K byte pages which are re-directed into the SDRAM address space (see Figure 8-1: “Memory Access Example using Direct Addressing (16M Byte Example)” on page 122.

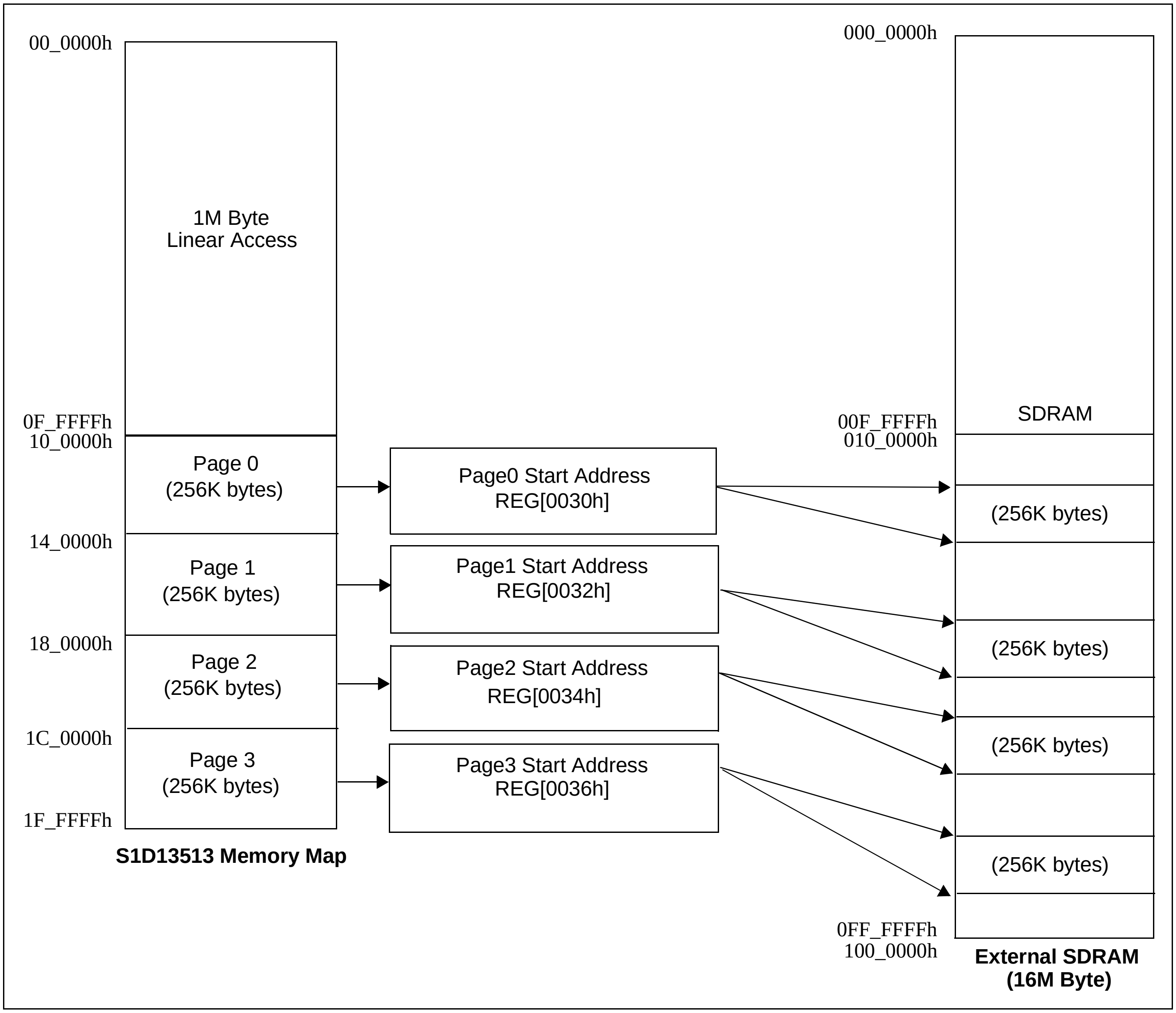


Figure 8-1: Memory Access Example using Direct Addressing (16M Byte Example)

8.2 Accessing Memory using Indirect Addressing

When indirect addressing is selected (see Section 5.3, “Summary of Configuration Options” on page 41), the SDRAM memory is addressed according to the address set in REG[0012h] ~ REG[0014h]. The address size allows access to the full range of possible SDRAM memory sizes. Once the address is set, the memory can be read/written through the Indirect Interface Memory Access Data Port, REG[0018h]. When a memory access completes, the address is incremented automatically.

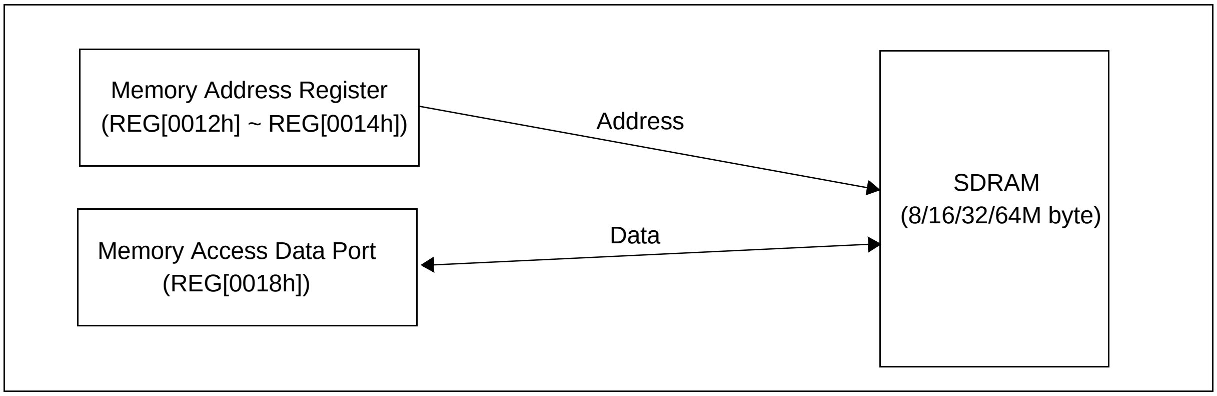


Figure 8-2: External Memory (Indirect Access)

8.3 IO Map

To access the functional registers for each IO device, refer to Table 10-2:, “S1D13513 Register Mapping,” on page 127.

9 Clocks

9.1 Clock Overview

The following diagram provides a logical representation of the S1D13513 internal clocks.

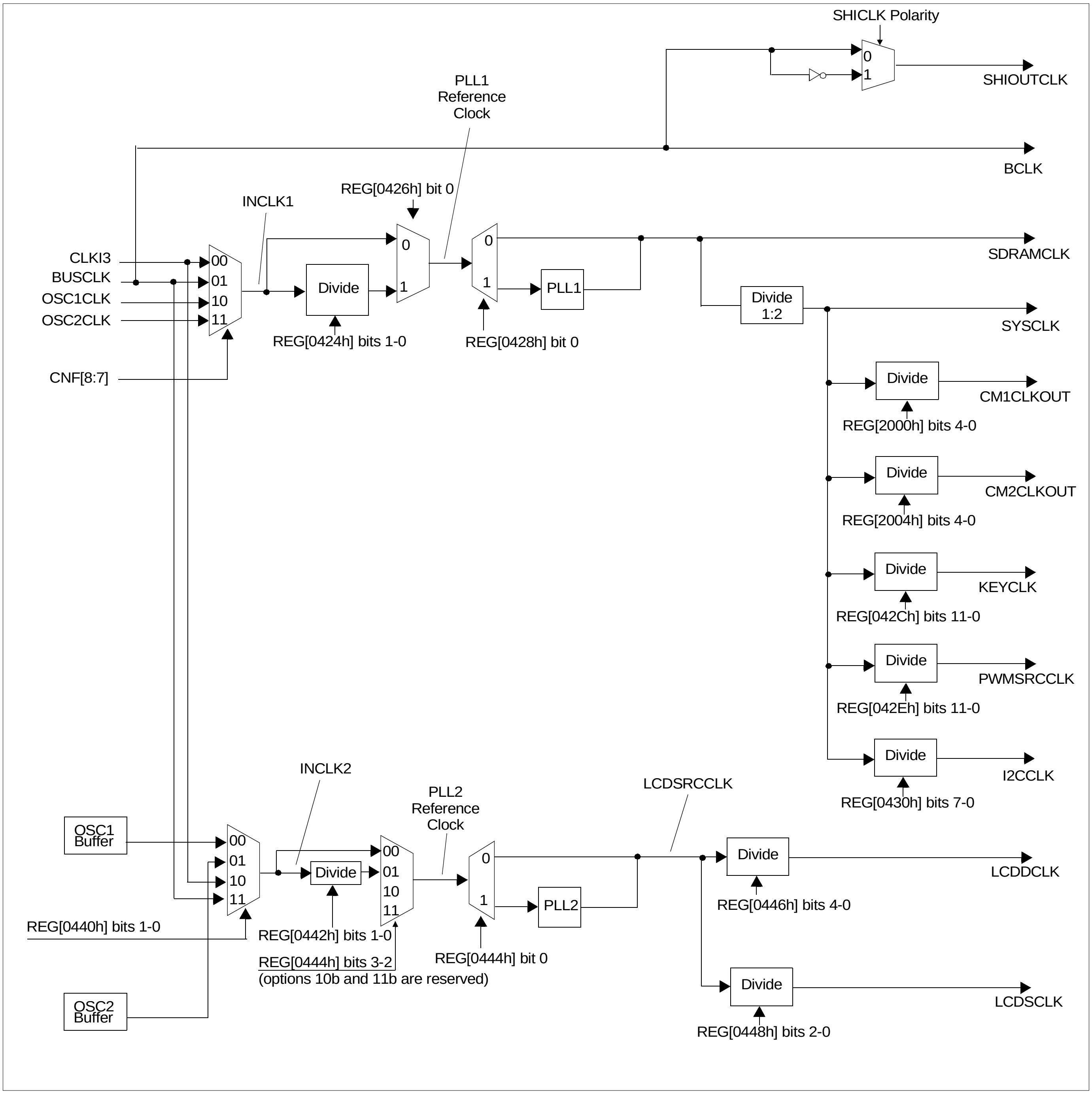


Figure 9-1: Clock Diagram

Note

If a host interface that uses BUSCLK is selected (CNF6=0), BUSCLK must remain on continuously while the S1D13513 is in normal operation mode.

9.2 PLL Programming Examples

Before changing the frequency of either PLL1 or PLL2, the PLL must be disabled using the PLL1 Enable bit (REG[0410h] bit 0) or PLL2 Enable bit (REG[0418h], as appropriate. Note that when the PLL is enabled after the changes, it is unstable for a maximum of 200µs.

Example 1: Set PLL2 for a fPLLOUT = 20MHz using fPLLREFCLK = 5MHz.

1. Find the value of NN.
   NN = fPLL2OUT ÷ fPLL2REFCLK
   NN = 20MHz ÷ 5MHz
   NN = 4
   Set the PLL2 N Multiplier bits 3-0 = 0011b (see REG[0414h] bits 3-0).

2. Find the VCO frequency.
   fVCO = fPLL2OUT x VV
   fVCO = 20 x 2
   fVCO = 40 MHz
   In this case, the default VV of 2 does not meet the following condition:
   100MHz <= fVCO <= 400MHz

   A VV of 8 does meet the condition (20MHz x 8 = 160MHz).
   Therefore, set the V Divider bits 1-0 to 11b (see REG[0414h] bits 5-4).

3. Set the PLL2 VC bits 3-0 for fVCO = 160MHz (REG[0414h] bits 11-8 = 0010b).

4. Set the Low Pass Filter resistance for a fPLL2REFCLK of 5MHz.
   Set the PLL2 RS bits 3-0 to 1010b (see REG[0414h] bits 15-12).

10 Registers

This section discusses how and where to access the S1D13513 registers. It also provides detailed information about the layout and usage of each register.

Note

If a Big Endian Host interface is selected, the registers must be accessed using the procedure shown in Section 20.6, "Register Accesses for Big Endian Host Interfaces" on page 452.

10.1 Register Mapping

The S1D13513 registers are memory-mapped. When the system decodes the input pins as CS# = 0 and M/R# = 0, the registers may be accessed.

Table 10-1 : S1D13513 Memory/Register Selection

M/R#	Address	Size	Function
1	000000h to 1FFFFFh	2M bytes	SDRAM memory Space
0	000000h to 1FFFFFh	2M bytes	All registers spaces

Note

When Power Save Mode is enabled, synchronous registers and SDRAM memory must not be accessed.

The register space is decoded by AB[20:0] and is mapped as follows.

Table 10-2: S1D13513 Register Mapping

Address	Type	Function
0000h to 0004h	Asynchronous	Host Interface Registers
0006h to 0044h	Synchronous	
0400h to 0472h	Asynchronous	System Control Registers
0800h to 081Ch	Synchronous	LCD Panel Configuration Registers
081Eh to 082Eh	Synchronous	HR-TFT Configuration Registers
0830h to 0870h	Synchronous	LCD Display Mode Registers
0C00h to 0C2Ah	Asynchronous	GPIO Registers
1000h to 14FEh	Synchronous	Sprite Registers
1700h to 179Eh	Synchronous	Sprite Engine Registers
1800h to 1AFEh	Synchronous	2D BitBLT Registers
1C00h to 1C14h	Synchronous	Memory Controller Registers
2000h to 2046h	Synchronous	Camera Interface Registers
2430h to 246Eh	Synchronous	Resizer Operation Registers
2800h to 2876h	Synchronous	YUV Capture Module Registers
3000h to 3014h	Synchronous	YRC Registers
3400h to 3412h	Synchronous	PWM Registers
3800h to 38FEh	Synchronous	I2C Registers
3C00h to 3C70h	Synchronous	DMA Control Registers
4000h to 4FFEh	Synchronous	Command FIFO (For BitBLT and Sprite) Registers
5000h to 500Eh	Synchronous	Keypad Interface Register

10.2 Register Set

The S1D13513 registers are listed in the following table.

Table 10-3: S1D13513 Register Set

Register	Page	Register	Page
Host Interface Registers			
REG[0000h] Product ID Register 0	133	REG[0002h] Product ID Register 1	133
REG[0004h] Embedded Memory Size Register	133	REG[0006h] through REG[0010h] are Reserved	134
REG[0012h] Indirect Interface Memory Address Register 0	134	REG[0014h] Indirect Interface Memory Address Register 1	134
REG[0018h] Indirect Interface Memory Access Data Port Register	135	REG[001Ch] through REG[001Eh] are Reserved	135
REG[0020h] Interrupt Status Register	136	REG[0022h] Interrupt Control Register	138
REG[0024h] Host Time-out Control Register	140	REG[0026h] Bus Error Interrupt Status Register	141
REG[0028h] Bus Error Interrupt Control Register	141	REG[002Ah] Interrupt Pin Control Register	142
REG[002Ch] through REG[002Eh] are Reserved	143	REG[0030h] SDRAM Host Page 0 Start Address Register	143
REG[0032h] SDRAM Host Page 1 Start Address Register	144	REG[0034h] SDRAM Host Page 2 Start Address Register	144
REG[0036h] SDRAM Host Page 3 Start Address Register	144	REG[0038h] through REG[0042h] are Reserved	145
REG[0044h] Host Configuration Register	145		
System Control Registers			
REG[0400h] through REG[0404h] are Reserved	146	REG[0406h] Configuration Pins Status Register	146
REG[0408h] OSC1 Control Register	146	REG[040Ah] OSC2 Control Register	147
REG[040Ch] PLL1 Configuration Register 0	147	REG[040Eh] PLL1 Configuration Register 1	149
REG[0410h] PLL1 Control Register	149	REG[0412h] is Reserved	149
REG[0414h] PLL2 Configuration Register 0	150	REG[0416h] PLL2 Configuration Register 1	151
REG[0418h] PLL2 Control Register	152	REG[041Ah] through REG[0422h] are Reserved	152
REG[0424h] PLL1 Reference Clock Divide Select Register	152	REG[0426h] PLL1 Control Register 0	153
REG[0428h] PLL1 Control Register 1	153	REG[042Ch] Key Clock Control Register	154
REG[042Eh] PWM Source Clock Control Register	155	REG[0430h] I2C Clock Control Register	156
REG[0440h] PLL2 Control Register 0	156	REG[0442h] PLL2 Control Register 1	157
REG[0444h] PLL2 Control Register 2	157	REG[0446h] LCD Clock Control Register 0	159
REG[0448h] LCD Clock Control Register 1	160	REG[0460h] Software Reset Register	161
REG[0462h] Clock Enable Register	161	REG[0464h] GPIOC&D Pull-down Resistor Control Register	162
REG[0466h] is Reserved	162	REG[0468h] GPIOG&H Pull-down Resistor Control Register	163
REG[046Ah] MEMDQ Pull-down Resistor Control Register 0	163	REG[046Ch] MEMDQ Pull-down Resistor Control Register 1	163
REG[046Eh] CNF Pull-down Resistor Control Register	164	REG[0470h] Power Down Mode Control Register	164
REG[0472h] Bus Time-out Reset Control Register	165	REG[04A0h] through REG[04A2h] are Reserved	165
LCD Panel Configuration Registers			
REG[0800h] LCD Panel Type Select Register	166	REG[0802h] LCD Horizontal Total Register	169
REG[0804h] LCD Horizontal Display Period Register	169	REG[0806h] LCD Horizontal Display Period Start Position Register	170
REG[0808h] LCD Horizontal Pulse Width Register	170	REG[080Ah] LCD Horizontal Pulse Start Position Register	171
REG[080Ch] LCD Vertical Total Register	171	REG[080Eh] LCD Vertical Display Period Register	171
REG[0810h] LCD Vertical Display Period Start Position Register	172	REG[0812h] LCD Vertical Pulse Width Register	172
REG[0814h] Vertical Pulse Start Position Register	172	REG[0816h] LCD Serial Interface Configuration Register	173
REG[0818h] LCD Status Register	174	REG[081Ah] LCD VSYNC Interrupt Delay Register	175
REG[081Ch] LCD Serial Command/Parameter Register	175	REG[081Eh] MOD/Serial Command Register	176
HR-TFT Configuration Registers			
REG[0820h] HR-TFT Configuration Register	177	REG[0822h] HR-TFT CLS Width Register	177
REG[0824] HR-TFT PS1 Rising Edge Register	178	REG[0826h] HR-TFT PS2 Rising Edge Register	178

Table 10-3: S1D13513 Register Set (Continued)

Register	Page	Register	Page
REG[0828h] HR-TFT PS2 Toggle Width Register	178	REG[082Ah] HR-TFT PS3 Signal Width Register	178
REG[082Ch] HR-TFT REV Toggle Point Register	179	REG[082Eh] HR-TFT PS1/2 End Register	179
LCD Display Mode Registers			
REG[0830h] Display Mode Setting Register 0	180	REG[0832h] Display Mode Setting Register 1	182
REG[0834h] Display Mode Setting Register 2	185	REG[0836h] PIP2 Window Alpha Blending Mode Register	188
REG[0838h] PIP2 Window Transparent Key Color Red Register 190		REG[083Ah] PIP2 Window Transparent Key Color Green Register 190	
REG[083Ch] PIP2 Window Transparent Key Color Blue Register 190		REG[083Eh] Gamma Control Register	191
REG[0840h] Gamma LUT Access Address Port Register	194	REG[0842h] Gamma LUT Access Data Port Register	194
REG[0844h] Pseudo Color Mode Register	195	REG[0846h] Display FIFO1 Threshold Register	196
REG[0848h] Display FIFO2 Threshold Register	196	REG[084Ah] PIP1 Window X Start Position Register	196
REG[084Ch] PIP1 Window X End Position Register	197	REG[084Eh] PIP1 Window Y Start Position Register	198
REG[0850h] PIP1 Window Y End Position Register	198	REG[0852h] PIP2 Window X Start Position Register	198
REG[0854h] PIP2 Window X End Position Register	199	REG[0856h] PIP2 Window Y Start Position Register	199
REG[0858h] PIP2 Window Y End Position Register	199	REG[085Ah] Main Window Front Buffer Start Address Register 0 200	
REG[085Ch] Main Window Front Buffer Start Address Register 1 200		REG[085Eh] PIP1 Window Front Buffer Start Address Register 0 200	
REG[0860h] PIP1 Window Front Buffer Start Address Register 1 200		REG[0862h] PIP2 Window Front Buffer Start Address Register 0 201	
REG[0864h] PIP2 Window Front Buffer Start Address Register 1 201		REG[0866h] Main/PIP1/PIP2 Window Back Buffer Start Address Register 0	201
REG[0868h] Main/PIP1/PIP2 Window Back Buffer Start Address Register 1	201	REG[086Ah] Main Window Front Buffer Line Address Offset Register 202	
REG[086Ch] PIP1 Window Front Buffer Line Address Offset Register 202		REG[086Eh] PIP2 Window Front Buffer Line Address Offset Register 202	
REG[0870h] Main/PIP1/PIP2 Window Back Buffer Line Address Offset Register	202	REG[0880h] Color Conversion Control Register	203
REG[0882h] ~ REG[0892h] Color Conversion Matrix Coefficient Registers 0-8	203		
GPIO Registers			
REG[0C00h] GPIOA Data Register	204	REG[0C02h] GPIOA Pin Function Register	205
REG[0C04h] GPIOB Data Register	206	REG[0C06h] GPIOB Pin Function Register	206
REG[0C08h] GPIOC Data Register	207	REG[0C0Ah] GPIOC Pin Function Register	207
REG[0C0Ch] GPIOD Data Register	208	REG[0C0Eh] GPIOD Pin Function Register	208
REG[0C10h] through REG[0C16h] are Reserved	208	REG[0C18h] GPIOG Data Register	209
REG[0C1Ah] GPIOG Pin Function Register	209	REG[0C1Ch] GPIOH Data Register	210
REG[0C1Eh] GPIOH Pin Function Register	210	REG[0C20h] through REG[0C22h] are Reserved	210
REG[0C24h] GPIOA&B Interrupt Type Register	211	REG[0C26h] GPIOA&B Interrupt Polarity Register	212
REG[0C28h] GPIOA&B Interrupt Enable Register	213	REG[0C2Ah] GPIOA&B IRQ Status and Clear Register	214
Sprite Registers when Sprite Engine is Enabled			
REG[1xxxh +00h] Sprite #n General Control Register	216	REG[1xxxh +02h] is Reserved	218
REG[1xxxh +04h] Sprite #n Image Start Address Register 0	218	REG[1xxxh +06h] Sprite #n Image Start Address Register 1	218
REG[1xxxh +08h] Sprite #n Rotated Image Start Address Register 0 219		REG[1xxxh +0Ah] Sprite #n Rotated Image Start Address Register 1 219	
REG[1xxxh +0Ch] Sprite #n X Position Register	219	REG[1xxxh +0Eh] Sprite #n Y Position Register	220
REG[1xxxh +10h] Sprite #n Frame Width Register	220	REG[1xxxh +12h] Sprite #n Frame Height Register	221
REG[1xxxh +14h] Sprite #n Reference Point X Offset Register	221	REG[1xxxh +16h] Sprite #n Reference Point Y Offset Register	222
REG[1xxxh +18h] Sprite #n Transparency Color/Texture Alpha Register 222		REG[1xxxh +1Ah] Sprite #n Color Format Register	223
REG[1xxxh +1Ch] Sprite #n Frame Sequence Register 0	224	REG[1xxxh +1Eh] Sprite #n Frame Sequence Register 1	224
REG[1xxxh +20h] Sprite #n Frame Sequence Register 2	224	REG[1xxxh +22h] Sprite #n Frame Sequence Register 3	224

Table 10-3: S1D13513 Register Set (Continued)

Register	Page	Register	Page
REG[1xxxh +24h] Sprite #n Frame Sequence Register 4	224	REG[1xxxh +26h] Sprite #n Frame Sequence Register 5	224
REG[1xxxh +28h] Sprite #n Frame Sequence Register 6	224	REG[1xxxh +2Ah] Sprite #n Frame Sequence Register 7	224
REG[1xxxh +2Ch] Sprite #n Virtual Image Width Register	226	REG[1xxxh +2Eh] Sprite #n Virtual Image Height Register	226
REG[1xxxh +30h] Sprite #n X Scan Vector H Register 0	227	REG[1xxxh +32h] Sprite #n X Scan Vector H Register 1	227
REG[1xxxh +34h] Sprite #n Y Scan Vector H Register 0	227	REG[1xxxh +36h] Sprite #n Y Scan Vector H Register 1	227
REG[1xxxh +38h] Sprite #n X Scan Vector V Register 0	228	REG[1xxxh +3Ah] Sprite #n X Scan Vector V Register 1	228
REG[1xxxh +3Ch] Sprite #n Y Scan Vector V Register 0	228	REG[1xxxh +3Eh] Sprite #n Y Scan Vector V Register 1	228
REG[1xxxh +40h] Sprite #n X Scan Offset Register 0	229	REG[1xxxh +42h] Sprite #n X Scan Offset Register 1	229
REG[1xxxh +44h] Sprite #n Y Scan Offset Register 0	229	REG[1xxxh +46h] Sprite #n Y Scan Offset Register 1	229
REG[1xxxh +48h] through REG[1xxxh +5Eh] are Reserved	230		
Sprite Registers when Image Format Converter is Enabled			
REG[1004h] IFC Source Image Address Register 0	231	REG[1006h] IFC Source Image Address Register 1	231
IFC Destination Image Address (see REG[1710h] ~ REG[1712h])			231
REG[102Ch] IFC Image Width Register	232	REG[102Eh] IFC Image Height Register	232
Sprite Engine Registers			
REG[1700h] Sprite Control Register	233	REG[1702h] Sprite Status Register	235
REG[1704h] Sprite Frame Sequence Trigger Control Register	236	REG[1706h] Sprite Interrupt Control Register	238
REG[1708h] Sprite Interrupt Status Register	238	REG[1710h] Sprite Frame Buffer 0 Start Address Register 0	239
REG[1712h] Sprite Frame Buffer 0 Start Address Register 1	239	REG[1714h] Sprite Frame Buffer 1 Start Address Register 0	240
REG[1716h] Sprite Frame Buffer 1 Start Address Register 1	240	REG[1718h] is Reserved	240
REG[1750h] through REG[1774h] are Reserved	240	REG[1780h] ~ REG[179Eh] Sprite #0-15 Frame Sequence Control Registers	241
2D BitBLT Registers			
REG[1800h] BitBLT Control Register 0	243	REG[1802h] BitBLT Control Register 1	243
REG[1804h] BitBLT Control Register 2	244	REG[1806h] is Reserved	245
REG[1808h] BitBLT Command Register	245	REG[180Ah] BitBLT Raster Operation Code Register	248
REG[1810h] BitBLT Source Base Address Register 0	249	REG[1812h] BitBLT Source Base Address Register 1	249
REG[1814h] BitBLT Source X Start Position Register	250	REG[1816h] BitBLT Source Y Start Position Register	251
REG[1818h] BitBLT Destination Base Address Register 0	252	REG[181Ah] BitBLT Destination Base Address Register 1	252
REG[181Ch] BitBLT Destination X Start Position Register	253	REG[181Eh] BitBLT Destination Y Start Position Register	254
REG[1820h] BitBLT Pattern Start Address Register 0	255	REG[1822h] BitBLT Pattern Start Address Register 1	255
REG[1824h] BitBLT Memory Address Offset Register	255	REG[1826h] BitBLT Width Register	256
REG[1828h] BitBLT Height Register	256	REG[1834h] BitBLT Clipping X Start Position Register	257
REG[1836h] BitBLT Clipping Y Start Position Register	258	REG[1838h] BitBLT Clipping Width Register	258
REG[1840h] BitBLT Clipping Height Register	259	REG[1842h] BitBLT Clipping Status Register	259
REG[1850h] BitBLT Background Color Register 0	260	REG[1852h] BitBLT Background Color Register 1	260
REG[1854h] BitBLT Foreground Color Register 0	260	REG[1856h] BitBLT Foreground Color Register 1	260
REG[1860h] BitBLT Color Expansion Start Position Register	261	REG[1862h] BitBLT Color Expansion Bit Format Register	262
REG[1870h] BitBLT Alpha Blending Source Format Register	263	REG[1872h] BitBLT Constant Alpha Register	263
REG[1874h] BitBLT Alpha Value Selection Register	264	REG[1876h] BitBLT Alpha Combine Alpha Map Register	265
REG[1880h] BitBLT Interrupt Status Register	265	REG[1882h] BitBLT Interrupt Control Register	266
REG[1886h] is Reserved	266	REG[1890h] BitBLT FIFO Status Register 0	266
REG[1892h] BitBLT FIFO Status Register 1	266	REG[1894h] BitBLT FIFO Status Register 2	267
REG[1896h] BitBLT FIFO Data Port Register	267	REG[1900h] ~ REG[1AFEh] BitBLT Color Expansion LUT Data Registers	267

Table 10-3: S1D13513 Register Set (Continued)

Register	Page	Register	Page
Memory Controller Registers			
REG[1C00h] Memory Control Register	268	REG[1C02h] Memory Configuration Register 0	268
REG[1C04h] Memory Configuration Register 1	269	REG[1C06h] Memory Configuration Register 2	271
REG[1C08h] Memory Advanced Configuration Register	273	REG[1C0Ah] Memory Initialization Configuration Register	274
REG[1C0Ch] Memory Refresh Timer Register	275	REG[1C0Eh] is Reserved	275
REG[1C10h] SDRAM Mode Setting Value Register	275	REG[1C12h] Mobile SDRAM Configuration Register	276
REG[1C14h] Mobile SDRAM Extended Mode Setting Register	277		
Camera Interface Registers			
REG[2000h] Camera1 Clock Setting Register	278	REG[2002h] Camera1 Signal Setting Register	279
REG[2004h] Camera2 Clock Setting Register	281	REG[2006h] Camera2 Signal Setting Register	282
REG[2008h] through REG[200Eh] are Reserved	283	REG[2010h] Camera Mode Setting Register	283
REG[2012h] Camera Frame Setting Register	285	REG[2014h] Camera Control Register	286
REG[2016h] Camera Status Register	288	REG[2020h] Strobe Control Signal Output Delay Setting Register	290
REG[2022h] Strobe Control Signal Pulse Width Setting Register	290	REG[2024h] Strobe Setting Register	290
REG[2028h] through REG[202Eh] are Reserved	292		
Resizer Operation Registers			
REG[2430h] Global Resizer Control Register	293	REG[2432h] through REG[243Eh] are Reserved	294
REG[2440h] View Resizer Control Register	294	REG[2444h] View Resizer Start X Position Register	295
REG[2446h] View Resizer Start Y Position Register	295	REG[2448h] View Resizer End X Position Register	296
REG[244Ah] View Resizer End Y Position Register	296	REG[244Ch] View Resizer Scaling Rate Register	297
REG[244Eh] View Resizer Scaling Mode Register	299	REG[2460h] Capture Resizer Control Register	299
REG[2464h] Capture Resizer Start X Position Register	300	REG[2466h] Capture Resizer Start Y Position Register	301
REG[2468h] Capture Resizer End X Position Register	301	REG[246Ah] Capture Resizer End Y Position Register	301
REG[246Ch] Capture Resizer Scaling Rate Register	302	REG[246Eh] Capture Resizer Scaling Mode Register	304
YUV Capture Module Registers			
REG[2800h] YUV Capture Control Register	305	REG[2802h] YUV Capture Status Flag Register	306
REG[2804h] YUV Capture Raw Status Flag Register	307	REG[2806h] YUV Capture Interrupt Control Register	309
REG[2808h] is Reserved	309	REG[280Ah] YUV Capture Start/Stop Control Register	310
REG[280Ch] through REG[280Eh] are Reserved	310		
YUV Capture FIFO Registers			
REG[2820h] YUV Capture FIFO Control Register	311	REG[2822h] YUV Capture FIFO Status Register	313
REG[2824h] YUV Capture FIFO Size Register	314	REG[2826h] YUV Capture FIFO Read/Write Port Register	314
REG[2828h] YUV Capture FIFO Valid Data Size Register	315	REG[282Ah] through REG[282Ch] are Reserved	315
REG[282Eh] YUV Capture FIFO Extend Register	315	REG[2830h] through REG[2870h] are Reserved	315
REG[2872h] YUV Horizontal Size Register	315	REG[2874h] YUV Vertical Size Register	316
REG[2876h] is Reserved	316		
YRC Registers			
REG[3000h] YRC Translate Mode Register	317	REG[3002h] YRC Write Start Address 0 Register 0	322
REG[3004h] YRC Write Start Address 0 Register 1	322	REG[3006h] YRC Write Start Address 1 Register 0	322
REG[3008h] YRC Write Start Address 1 Register 1	322	REG[300Ah] through REG[300Ch] Reserved	322
REG[300Eh] YRC UV Data Fix Register	323	REG[3010h] YRC Rectangular Pixel Width Register	323
REG[3012h] YRC Rectangular Line Address Offset Register	323	REG[3014h] YRC Memory Configuration Register	324

Table 10-3: S1D13513 Register Set (Continued)

Register	Page	Register	Page
PWM Registers			
REG[3400h] PWM Control Register	325	REG[3402h] PWM Clock Divide Register	328
REG[3404h] Red On/Off Control Register	328	REG[3406h] Green On/Off Control Register	329
REG[3408h] Blue On/Off Control Register	329	REG[340Ah] PWM Slope Register	330
REG[340Ch] PWM Duty Cycle Register	331	REG[340Eh] White LED Control Register	332
REG[3410h] through REG[3412h] are Reserved	332		
I2C Registers			
REG[3800h] I2C Control Register	333	REG[3804h] I2C Target Address Register	334
REG[3808h] is Reserved	334	REG[3810h] I2C Receive/Transmit Data Buffer and Command Register 335	
REG[3814h] I2C Standard Speed I2C Clock SCL High Count Register 335		REG[3818h] I2C Standard Speed I2C Clock SCL Low Count Register 336	
REG[381Ch] I2C Fast Speed I2C Clock SCL High Count Register 337		REG[3820h] I2C Fast Speed I2C Clock SCL Low Count Register 338	
REG[3824h] through REG[3828h] are Reserved	338	REG[382Ch] I2C Interrupt Status Register	339
REG[3830h] I2C Interrupt Enable Register	342	REG[3834h] I2C Interrupt Raw Status Register	343
REG[3838h] I2C Receive FIFO Threshold Register	347	REG[383Ch] I2C Transmit FIFO Threshold Register	347
REG[3840h] I2C Clear Combined and Individual Interrupt Register 347		REG[3844h] I2C Receive FIFO Underflow Interrupt Clear Register 348	
REG[3848h] I2C Receive FIFO Overflow Interrupt Clear Register 348		REG[384Ch] I2C Transmit FIFO Overflow Interrupt Clear Register 348	
REG[3850h] is Reserved	348	REG[3854h] I2C Transmit Abort Interrupt Clear Register	349
REG[3858h] is Reserved	349	REG[385Ch] I2C Busy Interrupt Clear Register	349
REG[3860h] I2C Stop Interrupt Clear Register	350	REG[3864h] I2C Start Interrupt Clear Register	350
REG[3868h] is Reserved	350	REG[386Ch] I2C Enable Register	351
REG[3870h] I2C Status Register	351	REG[3874h] I2C Transmit FIFO Level Register	352
REG[3878h] I2C Receive FIFO Level Register	353	REG[3880h] I2C Transmit Abort Source Register	353
REG[3888h] through REG[3890h] are Reserved	355	REG[38F4h] through REG[38FEh] are Reserved	355
DMA Control Registers			
REG[3C00h] DMA Channel 0 Source Address Register 0	356	REG[3C02h] DMA Channel 0 Source Address Register 1	356
REG[3C04h] DMA Channel 0 Destination Address Register 0	356	REG[3C06h] DMA Channel 0 Destination Address Register 1	356
REG[3C08h] DMA Channel 0 Transfer Count Register 0	357	REG[3C0Ah] DMA Channel 0 Transfer Count Register 1	357
REG[3C0Ch] DMA Channel 0 Control Register 0	357	REG[3C0Eh] DMA Channel 0 Control Register 1	359
REG[3C10h] DMA Channel 1 Source Address Register 0	360	REG[3C12h] DMA Channel 1 Source Address Register 1	360
REG[3C14h] DMA Channel 1 Destination Address Register 0	361	REG[3C16h] DMA Channel 1 Destination Address Register 1	361
REG[3C18h] DMA Channel 1 Transfer Count Register 0	361	REG[3C1Ah] DMA Channel 1 Transfer Count Register 1	361
REG[3C1Ch] DMA Channel 1 Control Register 0	362	REG[3C1Eh] DMA Channel 1 Control Register 1	364
REG[3C60h] DMA Channel Operating Select Register	365	REG[3C64h] DMA Channel Miscellaneous Register	365
REG[3C70h] DMA Channel Transfer Complete Control Register 366			
Command FIFO (For BitBLT and Sprite) Registers			
Command FIFO Registers mirror the offset of the Sprite and BitBLT registers (REG[1000h] ~ REG[17FFh] and REG[1800h] ~ REG[1FFFh]) but starting from a Base Address of REG[4000h]. For further information, refer to Section 10.4.19, "Command FIFO (For BitBLT and Sprite) Registers" on page 367.			
Keypad Interface Register			
REG[5000h] Key Control Register	368	REG[5002h] Key Interface Interrupt Status Register	369
REG[5004h] Key Scan Data Register	370	REG[5006h] Key Scan Input Filter Clock Register	370
REG[5008h] Key GPI Control Register	370	REG[500Ah] Key Scan Output Control Register	371
REG[500Ch] Key Scan GPI Filtered Register	372	REG[500Eh] Key Scan Re-Enable Register	372

10.3 Register Restrictions

All reserved bits must be set to 0b unless otherwise specified. Writing a value to a reserved bit may produce undefined results. Bits marked as n/a have no hardware effect. All register accesses must be 16-bit accesses.

10.4 Register Descriptions

10.4.1 Host Interface Registers

REG[0000h] Product ID Register 0 Default = 0200h							Read Only
Revision Code bits 7-0							
15	14	13	12	11	10	9	8
Reserved							
7	6	5	4	3	2	1	0

bits 15-8 Revision Code bits [7:0]
These bits indicate the revision code.
The revision code for the S1D13513 starts from 00h. 00h indicates first step, 01h indicates second step, and so on.

bits 7-0 Reserved
For the S1D13513 these bits always return 0000_0000b (00h).

REG[0002h] Product ID Register 1 Default = 002Ch							Read Only
Product Code bits 15-8							
15	14	13	12	11	10	9	8
Product Code bits 7-0							
7	6	5	4	3	2	1	0

bits 15-0 Product Code bits [15:0]
These bits indicate the product code.
The product code for the S1D13513 is 002Ch.

REG[0004h] Embedded Memory Size Register Default = 0000h							Read Only
Embedded Memory Size bits 15-8							
15	14	13	12	11	10	9	8
Embedded Memory Size bits 7-0							
7	6	5	4	3	2	1	0

bits 15-0 Embedded Memory Size bits [15:0]
These bits indicate the size of the embedded memory. Since the S1D13513 has no embedded memory, these bits return 0000h.

REG[0006h] through REG[0010h] are Reserved

These registers are Reserved and should not be written.

REG[0012h] Indirect Interface Memory Address Register 0 Default = 0000h							Read/Write
Indirect Interface Memory Address bits 15-8							
15	14	13	12	11	10	9	8
Indirect Interface Memory Address bits 7-1							Indirect Interface Memory Access Read/Write Select
7	6	5	4	3	2	1	0

REG[0014h] Indirect Interface Memory Address Register 1 Default = 0000h							Read/Write
Reserved	Indirect Interface Memory Address bits 30-24						
15	14	13	12	11	10	9	8
Indirect Interface Memory Address bits 23-16							
7	6	5	4	3	2	1	0

REG[0012h] bit 0 Indirect Interface Memory Access Read/Write Select
This bit is used for Indirect Interface modes only.
This bit selects whether a memory access done through the Indirect Interface Memory Access Data Port (REG[0018h]) is a read or write access.
When this bit = 0b, a write access takes place. (default)
When this bit = 1b, a read access takes place.

REG[0014h] bit 15 Reserved
The default value for this bit is 0b.

REG[0014h] bits 14-0
REG[0012h] bits 15-1 Indirect Interface Memory Address bits [30:1]
These bits are used for Indirect Interface modes only.
These bits specify the address used for each memory access when an indirect interface is selected.
REG[0014h] bits 9-0, REG[0012h] bits 15-1
= Indirect Interface Memory Address bits 25-1

For further information on accessing memory using the indirect interface, see Section 8.2, “Accessing Memory using Indirect Addressing” on page 123.

REG[0016h] is Reserved

This register is Reserved and should not be written.

REG[0018h] Indirect Interface Memory Access Data Port Register							
Default = 0000h							Read/Write
Indirect Interface Memory Access Data Port bits 15-8							
15	14	13	12	11	10	9	8
Indirect Interface Memory Access Data Port bits 7-0							
7	6	5	4	3	2	1	0

bits 15-0

Indirect Interface Memory Access Data Port bits [15:0]
These bits are used for Indirect Interface modes only.
These bits are the memory read/write port for the Indirect Interface. For more information on using the Indirect Interface, see Section 20.3, “Indirect Interface” on page 438.

REG[001Ah] Memory Access Status Register							
Default = 0000h							Read Only
Memory Busy	n/a						
15	14	13	12	11	10	9	8
n/a							
7	6	5	4	3	2	1	0

bit 15

Memory Busy
When this bit = 0b, the memory controller is idle and the Host CPU can access memory. Data read / write is possible via the Indirect Interface Memory Access Data Port (REG[0018h]).
When this bit = 1b, the memory controller is busy and the Host CPU cannot access memory.

REG[001Ch] through REG[001Eh] are Reserved

These registers are Reserved and should not be written.

REG[0020h] Interrupt Status Register Default = 0000h							Read/Write
n/a				YUV Capture FIFO Interrupt Flag	Reserved	DMAC Interrupt Flag	I2C Interrupt Flag
15	14	13	12	11	10	9	8
Key Interrupt Flag	Camera Interrupt Flag	BitBLT Interrupt Flag	Sprite Interrupt Flag	GPIO Interrupt Flag	VSYNC Interrupt Flag	Reserved	Host Interrupt Flag
7	6	5	4	3	2	1	0

bit 11 YUV Capture FIFO Interrupt Flag
This bit indicates the status of the YUV FIFO interrupt which occurs when one of the YUV Capture FIFO flags is triggered (see REG[2802h] bits 10-8). This bit is not masked by the YUV Capture FIFO Interrupt Enable bit, REG[0022h] bit 11.
When this bit = 0b, a YUV Capture FIFO interrupt has not occurred.
When this bit = 1b, a YUV Capture FIFO interrupt has occurred.

To clear this flag, clear the YUV Capture FIFO flags in REG[2802h] bits 10-8.

bit 10 Reserved
The default value for this bit is 0b.

bit 9 DMAC Interrupt Flag
This bit indicates the status of the DMAC interrupt which occurs when a DMA channel 0 or DMA channel 1 transfer ends (see REG[3C0Ch] bit 1 or REG[3C1Ch] bit 1). This bit is not masked by the DMAC Interrupt Enable bit, REG[0022h] bit 9.
When this bit = 0b, a DMAC interrupt has not occurred.
When this bit = 1b, a DMAC interrupt has occurred.

To clear this flag, clear the appropriate DMA Channel Transfer End bit in either REG[3C0Ch] bit 1 or REG[3C1Ch] bit 1

bit 8 I2C Interrupt Flag
This bit indicates the status of the I2C interrupt which occurs when an I2C Interrupt Status Flag is triggered (see REG[382Ch]. This bit is not masked by the I2C Interrupt Enable bit, REG[0022h] bit 8.
When this bit = 0b, an I2C interrupt has not occurred.
When this bit = 1b, an I2C interrupt has occurred.

To clear this flag, clear all I2C Interrupt Status Flags in REG[382Ch] or REG[3834h].

bit 7 Key Interrupt Flag
This bit indicates the status of the Keypad Interface interrupt which occurs when a Key Interrupt Status Flag is triggered (see REG[5002h] and REG[500Eh]). This bit is not masked by the Key Interrupt Enable bit, REG[0022h] bit 7.
When this bit = 0b, a Key interrupt has not occurred.
When this bit = 1b, a Key interrupt has occurred.

To clear this flag, write a 1b to REG[500Eh] bit 0.

bit 6 Camera Interrupt Flag
This bit indicates the status of the Camera interrupt which occurs when the frame capture completes (see REG[2016h] bit 1). This bit is not masked by the Camera Interrupt Enable bit, REG[0022h] bit 6.
When this bit = 0b, a Camera interrupt has not occurred.
When this bit = 1b, a Camera interrupt has occurred.

To clear this flag, clear the Frame Capture Interrupt Status bit in REG[2014h] bit 1.

bit 5 BitBLT Interrupt Flag
This bit indicates the status of the BitBLT interrupt which occurs when the BitBLT operation completes (see REG[1880h] bit 0). This bit is not masked by the BitBLT Interrupt Enable bit, REG[0022h] bit 5.
When this bit = 0b, a BitBLT interrupt has not occurred.
When this bit = 1b, a BitBLT interrupt has occurred.

To clear this flag, clear the BitBLT Interrupt Status bit in REG[1880h] bit 0.

bit 4 Sprite Interrupt Flag
This bit indicates the status of the Sprite interrupt which occurs when the Sprite drawing process completes (see REG[1708h] bit 1). This bit is masked by the Sprite Interrupt Enable bit, REG[1706h] bit 1.
When this bit = 0b, a Sprite interrupt has not occurred.
When this bit = 1b, a Sprite interrupt has occurred.

To clear this flag, clear the Sprite Raw Interrupt Status bit in REG[1708h] bit 1.

bit 3 GPIO Interrupt Flag
This bit indicates the status of the GPIO interrupt which occurs when one of the GPIOA or GPIOB Interrupt Status bits returns a 1b (see REG[0C2Ah]). This bit is not masked by the GPIO Interrupt Enable bit, REG[0022h] bit 3.
When this bit = 0b, a GPIO interrupt has not occurred.
When this bit = 1b, a GPIO interrupt has occurred.

To clear this flag, clear the GPIO Interrupt Status bits in REG[0C2Ah].

bit 2 VSYNC Interrupt Flag
This bit indicates the status of the VSYNC interrupt which occurs when the VSYNC Interrupt Status bit returns a 1b (see REG[0818h] bit 11). This bit is not masked by the VSYNC Interrupt Enable bit, REG[0022h] bit 2.
When this bit = 0b, a VSYNC interrupt has not occurred.
When this bit = 1b, a VSYNC interrupt has occurred.

To clear this flag, clear the VSYNC Interrupt Status bit in REG[0818h] bit 11.

bit 1 Reserved
The default value for this bit is 0b.

bit 0 Host Interrupt Flag
This bit indicates the status of the Host interrupt which is generated when a Host Read/Write Cycle Time-out (see REG[0026h] bits 1-0) or Host WAIT# Length Time-out (see REG[0472h] bits 2-0) occurs. This bit is not masked by the Host Interrupt Enable bit, REG[0022h] bit 0. For further information on the time-out functions, refer to Section 20.2, “Host Bus Time-out Function” on page 436.
When this bit = 0b, a Host interrupt has not occurred.
When this bit = 1b, a Host interrupt has occurred.

To clear this flag, clear the interrupt flags in REG[0026h] bits 1-0.

REG[0022h] Interrupt Control Register
Default = 0000h Read/Write

n/a				YUV Capture FIFO Interrupt Enable	Reserved	DMAC Interrupt Enable	I2C Interrupt Enable
15	14	13	12	11	10	9	8
Key Interrupt Enable	Camera Interrupt Enable	BitBLT Interrupt Enable	Sprite Interrupt Enable	GPIO Interrupt Enable	VSYNC Interrupt Enable	Reserved	Host Interrupt Enable
7	6	5	4	3	2	1	0

bit 11 YUV Capture FIFO Interrupt Enable
This bit controls whether a YUV Capture FIFO interrupt causes an interrupt request on the INT2# pin. The status of the YUV Capture FIFO interrupt is indicated by the YUV Capture FIFO Interrupt Flag, REG[0020h] bit 11.
When this bit = 0b, an interrupt request is not generated.
When this bit = 1b, an interrupt request is generated.

bit 10 Reserved
The default value for this bit is 0b.

bit 9 DMAC Interrupt Enable
This bit controls whether a DMAC interrupt causes an interrupt request on the INT2# pin. The status of the DMAC interrupt is indicated by the DMAC Interrupt Flag, REG[0020h] bit 9.
When this bit = 0b, an interrupt request is not generated.
When this bit = 1b, an interrupt request is generated.

bit 8 I2C Interrupt Enable
This bit controls whether a I2C interrupt causes an interrupt request on the INT2# pin. The status of the I2C interrupt is indicated by the I2C Interrupt Flag, REG[0020h] bit 8.
When this bit = 0b, an interrupt request is not generated.
When this bit = 1b, an interrupt request is generated.

bit 7 Key Interrupt Enable
This bit controls whether a Keypad Interface interrupt causes an interrupt request on the INT2# pin. The status of the Key interrupt is indicated by the Key Interrupt Flag, REG[0020h] bit 7.
When this bit = 0b, an interrupt request is not generated.
When this bit = 1b, an interrupt request is generated.

bit 6 Camera Interrupt Enable
This bit controls whether a Camera interrupt causes an interrupt request on the INT2# pin. The status of the Camera interrupt is indicated by the Camera Interrupt Flag, REG[0020h] bit 6.
When this bit = 0b, an interrupt request is not generated.
When this bit = 1b, an interrupt request is generated.

bit 5 BitBLT Interrupt Enable
This bit controls whether a BitBLT interrupt causes an interrupt request on the INT2# pin. The status of the BitBLT interrupt is indicated by the BitBLT Interrupt Flag, REG[0020h] bit 5.
When this bit = 0b, an interrupt request is not generated.
When this bit = 1b, an interrupt request is generated.

bit 4 Sprite Interrupt Enable
This bit controls whether a Sprite interrupt causes an interrupt request on the INT2# pin. The status of the Sprite interrupt is indicated by the Sprite Interrupt Flag, REG[0020h] bit 4.
When this bit = 0b, an interrupt request is not generated.
When this bit = 1b, an interrupt request is generated.

bit 3 GPIO Interrupt Enable
This bit controls whether a GPIO interrupt causes an interrupt request on the INT2# pin. The status of the GPIO interrupt is indicated by the GPIO Interrupt Flag, REG[0020h] bit 3.
When this bit = 0b, an interrupt request is not generated.
When this bit = 1b, an interrupt request is generated.

bit 2 VSYNC Interrupt Enable
This bit controls whether a VSYNC interrupt causes an interrupt request on the INT2# pin. The status of the VSYNC interrupt is indicated by the VSYNC Interrupt Flag, REG[0020h] bit 2.
When this bit = 0b, an interrupt request is not generated.
When this bit = 1b, an interrupt request is generated.

bit 1 Reserved
The default value for this bit is 0b.

bit 0 Host Interface Interrupt Enable
This bit controls whether a Host interrupt causes an interrupt request on the INT2# pin. The status of the Host interrupt is indicated by the Host Interrupt Flag, REG[0020h] bit 0.
When this bit = 0b, an interrupt request is not generated.
When this bit = 1b, an interrupt request is generated.

REG[0024h] Host Time-out Control Register
Default = 007Fh Read/Write

n/a							
15	14	13	12	11	10	9	8
Host Time-out Enable	Host Time-out Value bits 6-0						
7	6	5	4	3	2	1	0

bit 7 Host Time-out Enable
This bit controls the bus time-out function for the host bus interface. A host bus time-out occurs when the host accesses SDRAM and the host cycle exceeds the time-out value specified by the Host Time-out Value bits, REG[0024h] bits 6-0. For further information, see Section 20.2, “Host Bus Time-out Function” on page 436.
When this bit = 0b, the time-out function is disabled.
When this bit = 1b, the time-out function is enabled.

bits 6-0 Host Time-out Value bits [6:0]
These bits specify the time-out value for the Host Time-out function, in system clocks.
When these bits are set to 00h, a Host Time-out will not occur.

REG[0026h] Bus Error Interrupt Status Register
Default = 0000h Read/Write

n/a						Reserved	Reserved
15	14	13	12	11	10	9	8
n/a						Memory Read Error Interrupt Flag	Memory Write Error Interrupt Flag
7	6	5	4	3	2	1	0

bit 9 Reserved
The default value for this bit is 0b.

bit 8 Reserved
The default value for this bit is 0b.

bit 1 Memory Read Error Interrupt Flag
This bit indicates the status of the Memory Read Error interrupt which occurs when a read access does not return the value within the specified time (see REG[0024h] bits 6-0). This bit is masked by the Memory Read Error Interrupt Enable bit and is only available when REG[0028h] bit 1 = 1b.
When this bit = 0b, a Memory Read Error interrupt has not occurred.
When this bit = 1b, a Memory Read Error interrupt has occurred.

bit 0 Memory Write Error Interrupt Flag
This bit indicates the status of the Memory Write Error interrupt which occurs when a write access does not complete within the specified time (see REG[0024h] bits 6-0). This bit is masked by the Memory Write Error Interrupt Enable bit and is only available when REG[0028h] bit 0 = 1b.
When this bit = 0b, an Memory Write Error interrupt has not occurred.
When this bit = 1b, an Memory Write Error interrupt has occurred.

REG[0028h] Bus Error Interrupt Control Register
Default = 0000h Read/Write

n/a						Reserved	Reserved
15	14	13	12	11	10	9	8
n/a						Memory Read Error Interrupt Enable	Memory Write Error Interrupt Enable
7	6	5	4	3	2	1	0

bit 9 Reserved
The default value for this bit is 0b.

bit 8 Reserved
The default value for this bit is 0b.

bit 1 Memory Read Error Interrupt Enable
This bit controls the Memory Read Error interrupt. The status is indicated by the Memory Read Error Interrupt Flag, REG[0026h] bit 1.
When this bit = 0b, the interrupt is disabled.
When this bit = 1b, the interrupt is enabled.

bit 0 Memory Write Error Interrupt Enable
This bit controls the Memory Write Error interrupt. The status is indicated by the Memory Write Error Interrupt Flag, REG[0026h] bit 0.
When this bit = 0b, the interrupt is disabled.
When this bit = 1b, the interrupt is enabled.

REG[002Ah] Interrupt Pin Control Register
Default = 0000h Read/Write

INT1# Pin Interrupt Enable	INT2# Output Control	INT1# Output Control	INT2# Pin Polarity Select	INT1# Pin Polarity Select	n/a		
15	14	13	12	11	10	9	8
n/a							
7	6	5	4	3	2	1	0

bit 15 INT1# Pin Interrupt Enable
This bit controls whether the INT1# pin sends interrupt requests to the Host. The INT1# pin is not masked by the interrupt enables in REG[0022h].
When this bit = 0b, the INT1# pin does not send interrupt requests to the Host (output is disabled).
When this bit = 1b, the INT1# pin sends interrupt requests to the Host (output is enabled).

bit 14 INT2# Output Control
This bit controls the output of the INT2# pin. For a summary, refer to Table 10-4, “INT[2:1]# Output Control/Pin Polarity Select Summary,” on page 143.
When this bit = 0b and the INT2# Pin Polarity Select bit is set for active low, INT2# is driven Hi-Z when inactive and LOW when active.
When this bit = 0b and the INT2# Pin Polarity Select bit is set for active high, INT2# is driven LOW when inactive and HIGH when active.
When this bit = 1b, the output is always driven HIGH or LOW depending on the INT2# polarity.

bit 13 INT1# Output Control
This bit controls the output of the INT1# pin. For a summary, refer to Table 10-4, “INT[2:1]# Output Control/Pin Polarity Select Summary,” on page 143.
When this bit = 0b and the INT1# Pin Polarity Select bit is set for active low, INT1# is driven Hi-Z when inactive and LOW when active.
When this bit = 0b and the INT1# Pin Polarity Select bit is set for active high, INT1# is driven LOW when inactive and HIGH when active.
When this bit = 1b, the output is always driven HIGH or LOW depending on the INT1# polarity.

bit 12 INT2# Pin Polarity Select
This bit selects the polarity of the INT2# pin. For a summary, refer to Table 10-4, “INT[2:1]# Output Control/Pin Polarity Select Summary,” on page 143.
When this bit = 0b, the INT2# pin is active low.
When this bit = 1b, the INT2# pin is active high.

bit 11 INT1# Pin Polarity Select
This bit selects the polarity of the INT1# pin. For a summary, refer to Table 10-4, “INT[2:1]# Output Control/Pin Polarity Select Summary,” on page 143.
When this bit = 0b, the INT1# pin is active low.
When this bit = 1b, the INT1# pin is active high.

Table 10-4 : INT[2:1]# Output Control/Pin Polarity Select Summary

REG[002Ah] bits 12-11	INT1#/INT2# Interrupt Status	REG[002Ah] bits 14-13	
		00b	**11b**
00b	Interrupt has not occurred	INT[2:1]# = HIGHZ	INT[2:1]# = Driven HIGH
	Interrupt has occurred	INT[2:1]# = Driven LOW	INT[2:1]# = Driven LOW
11b	Interrupt has not occurred	INT[2:1]# = Driven LOW	INT[2:1]# = Driven LOW
	Interrupt has occurred	INT[2:1]# = Driven HIGH	INT[2:1]# = Driven HIGH

REG[002Ch] through REG[002Eh] are Reserved

These registers are Reserved and should not be written.

REG[0030h] SDRAM Host Page 0 Start Address Register
Default = 0000h Read/Write

n/a						SDRAM Host Page 0 Start Address bits 25-24	
15	14	13	12	11	10	9	8
SDRAM Host Page 0 Start Address bits 23-18						n/a	
7	6	5	4	3	2	1	0

bits 9-2 SDRAM Host Page 0 Start Address bits [25:18]
These bits specify bits 25-18 of the memory start address for Page 0, which re-directs Host CPU memory accesses to a 256K byte window in SDRAM. Page 0 is accessed by the HOST CPU through memory address 100000h - 13FFFFh. For further information on accessing the external SDRAM memory, see Section 8.1, “Accessing Memory using Direct Addressing” on page 121.
REG[0030h] bits 9-2 = SDRAM Host Page 0 Start Address bits [25:18]

REG[0032h] SDRAM Host Page 1 Start Address Register Default = 0000h							Read/Write
n/a						SDRAM Host Page 1 Start Address bits 25-24	
15	14	13	12	11	10	9	8
SDRAM Host Page 1 Start Address bits 23-18						n/a	
7	6	5	4	3	2	1	0

bits 9-2 SDRAM Host Page 1 Start Address bits [25:18]
These bits specify bits 25-18 of the memory start address for Page 1, which re-directs Host CPU memory accesses to a 256K byte window in SDRAM. Page 1 is accessed by the HOST CPU through memory address 140000h - 17FFFFh. For further information on accessing the external SDRAM memory, see Section 8.1, “Accessing Memory using Direct Addressing” on page 121.
REG[0032h] bits 9-2 = Host SDRAM Page 1 Start Address bits [25:18]

REG[0034h] SDRAM Host Page 2 Start Address Register Default = 0000h							Read/Write
n/a						SDRAM Host Page 2 Start Address bits 25-24	
15	14	13	12	11	10	9	8
SDRAM Host Page 2 Start Address bits 23-18						n/a	
7	6	5	4	3	2	1	0

bits 9-2 SDRAM Host Page 2 Start Address bits [25:18]
These bits specify bits 25-18 of the memory start address for Page 2, which re-directs Host CPU memory accesses to a 256K byte window in SDRAM. Page 2 is accessed by the HOST CPU through memory address 180000h - 1BFFFFh. For further information on accessing the external SDRAM memory, see Section 8.1, “Accessing Memory using Direct Addressing” on page 121.
REG[0034h] bits 9-2 = Host SDRAM Page 2 Start Address bits [25:18]

REG[0036h] SDRAM Host Page 3 Start Address Register Default = 0000h							Read/Write
Reserved						SDRAM Host Page 3 Start Address bits 25-24	
15	14	13	12	11	10	9	8
SDRAM Host Page 3 Start Address bits 23-18						n/a	
7	6	5	4	3	2	1	0

bits 9-2 Host SDRAM Page 3 Start Address bits [25:18]
These bits specify bits 25-18 of the memory start address for Page 3, which re-directs Host CPU memory accesses to a 256K byte window in SDRAM. Page 3 is accessed by the HOST CPU through memory address 1C0000h - 1FFFFFh. For further information on accessing the external SDRAM memory, see Section 8.1, “Accessing Memory using Direct Addressing” on page 121.
REG[0036h] bits 7-0 = Host SDRAM Page 3 Start Address bits [25:18]

REG[0038h] through REG[0042h] are Reserved

These registers are Reserved and should not be written.

REG[0044h] Host Configuration Register
Default = 0000h Read/Write

Prefetch Buffer Disable	n/a				Reserved	n/a	
15	14	13	12	11	10	9	8
n/a				Reserved			Reserved
7	6	5	4	3	2	1	0

bit 15 Prefetch Buffer Disable
This bit enables/disables the prefetch buffer used for memory access acceleration. When the prefetch buffer is enabled, memory coherency issues may be possible (see Section 20.4, “Read Ahead Feature” on page 442).
When this bit = 0b, the prefetch buffer is enabled. (default)
When this bit = 1b, the prefetch buffer is disabled.

Note

1. The prefetch buffer must be enabled for indirect memory accesses.
2. This bit only applies for S1D13513 revision 02h. For revisions 00h and 01h, the prefetch buffer is always enabled regardless of the setting of this bit. The product revision can be checked by reading REG[0000h] bits 15-8.

bit 10 Reserved
This bit must set to 0b.

bits 3-1 Reserved
These bits must be set to 000b.

bit 0 Reserved
This bit must set to 0b.

10.4.2 System Control Registers

REG[0400h] through REG[0404h] are Reserved

These registers are Reserved and should not be written.

REG[0406h] Configuration Pins Status Register Default = xxxxh							Read Only
n/a							CNF8 Status
15	14	13	12	11	10	9	8
CNF[7:0] Status							
7	6	5	4	3	2	1	0

bits 8-0 CNF[8:0] Status bits (Read Only)
These bits indicate the current status of the corresponding S1D13513 configuration pins CNF[8:0]. The state of the CNF[8:0] pins only have an effect at the rising edge of RESET#. If the state of the CNF[8:0] pins changes after the rising edge of RESET#, it has no effect but the change will be indicated in this register. For a functional description of each configuration pin, see Section 5.3, "Summary of Configuration Options" on page 41.

REG[0408h] OSC1 Control Register Default = 0000h							Read/Write
n/a							
15	14	13	12	11	10	9	8
n/a							OSC1 Enable
7	6	5	4	3	2	1	0

bit 0 OSC1 Enable
This bit controls OSC1. A typical use for OSC1 is to provide the clock source for the panel interface when the PLL1 Source is not convenient. For a detailed diagram of the clock structure, refer to Figure 9-1: "Clock Diagram" on page 124.
When this bit = 0b, OSC1 is disabled. (default)
When this bit = 1b, OSC1 is enabled.

Note

1. If OSC1 is selected as the PLL2 source (REG[0440h] bits 1-0 = 00b), the LCD output must be disabled (REG[0830h] bit 0 = 0b) before disabling OSC1.
2. This bit is not affected by a software reset.
3. After enabling OSC1, wait until the oscillator becomes stable before performing the next register write. Note that the time required for the oscillator to become stable depends on the on-board crystal circuitry.

REG[040Ah] OSC2 Control Register Default = 0000h							Read/Write
n/a							
15	14	13	12	11	10	9	8
n/a							OSC2 Enable
7	6	5	4	3	2	1	0

bit 0

OSC2 Enable
This bit controls OSC2. A typical use for OSC2 is to provide a 27MHz clock source for YUV digital output. For a detailed diagram of the clock structure, refer to Figure 9-1: “Clock Diagram” on page 124.
When this bit = 0b, OSC2 is disabled. (default)
When this bit = 1b, OSC2 is enabled.

Note

1. If OSC2 is selected as the PLL2 source (REG[0440h] bits 1-0 = 01b), the LCD output must be disabled (REG[0830h] bit 0 = 0b) before disabling OSC2.
2. This bit is not affected by software reset.
3. After enabling OSC2, wait until the oscillator becomes stable before performing the next register write. Note that the time required for the oscillator to become stable depends on the on-board crystal circuitry.

REG[040Ch] PLL1 Configuration Register 0 Default = 0000h							Read/Write
PLL1 RS bits 3-0				PLL1 VC bits 3-0			
15	14	13	12	11	10	9	8
Reserved		PLL1 V Divider bits 1-0		PLL1 N Multiplier bits 3-0			
7	6	5	4	3	2	1	0

Note

For example, the value of 8311h results in a PLL1 output of 100MHz when a 50MHz reference clock is input to PLL1.

bits 15-12

PLL1 RS bits [3:0]
These bits are used to configure the Low Pass Filter (LPF) resistance and should be set based on the frequency of the PLL1 reference clock.

Table 10-5 : PLL1 RS Configuration

REG[040Ch] bits 15-12	PLL1 Reference Clock Frequency
0000b ~ 0111b	Reserved
1000b	20MHz ≤ fPLL1REFCLK ≤ 150MHz
1001b	Reserved
1010b	5MHz ≤ fPLL1REFCLK ≤ 20MHz
1011b ~ 1111b	Reserved

bits 11-8 PLL1 VC bits [3:0]

These bits set the analog adjustment pins for PLL1 and should be set according to the VCO frequency.

Table 10-6 : PLL1 VC Configuration

REG[040Ch] bits 11-8	PLL1 VCO Frequency
0000b	Reserved
0001b	100MHz ≤ fVCO ≤ 120MHz
0010b	120MHz < fVCO ≤ 160MHz
0011b	160MHz < fVCO ≤ 200MHz
0100b	200MHz < fVCO ≤ 240MHz
0101b	240MHz < fVCO ≤ 280MHz
0110b	280MHz < fVCO ≤ 320MHz
0111b	320MHz < fVCO ≤ 360MHz
1000b	360MHz < fVCO ≤ 400MHz
1001b ~ 1111b	Reserved

bits 7-6 Reserved

The default value for these bits is 00b.

bits 5-4 PLL1 V Divider bits [1:0]

These bits are used to configure the VCO frequency which must be set between 100MHz and 400MHz. These bits should be set using the following formula.

fVCO = fPLL1OUT x VV

Where:

fVCO is the frequency of VCO, in MHz
fPLL1OUT is the desired PLL1 output frequency, in MHz (see N Multiplier bits)
VV is the value based on the V Divider bits as follows.

Table 10-7 : VV Value

REG[040Ch] bits 5-4	VV Value
00b	Reserved
01b	2
10b	4
11b	8

Note

Normally VV is set to 2. When fPLL1OUT is lower than 50MHz, stabilize VCCO by setting VV to 4 or 8. Also, the PLL1 VC bits (REG[040Ch] bits 11-8) must be set according to the resulting fVCO. The frequency of VCO (fVCO) must always be within 100MHz ~ 400MHz.

bits 3-0 PLL1 N Multiplier bits [3:0]
These bits are used to determine the output frequency of PLL1 according to the following formula.

fPLL1OUT = fPLL1REFCLK x NN

Where:
fPLL1OUT is the desired PLL1 output frequency, in MHz
fPLL1REFCLK is the PLL1 reference clock input frequency, in MHz
NN is the N Multiplier value + 1

REG[040Eh] PLL1 Configuration Register 1
Default = 0000h Read/Write

PLL1 Configuration 1 bits 15-8							
15	14	13	12	11	10	9	8
PLL1 Configuration 1 bits 7-0							
7	6	5	4	3	2	1	0

bits 15-0 PLL1 Configuration 1 bits [15:0]
These bits are used to configure PLL1 and should be set to the recommended value of 0040h.

REG[0410h] PLL1 Control Register
Default = 0000h Read/Write

n/a							
15	14	13	12	11	10	9	8
n/a							PLL1 Enable
7	6	5	4	3	2	1	0

bit 0 PLL1 Enable
This bit controls PLL1. PLL1 must be disabled before changing the PLL1 Configuration registers, REG[040Ch] ~ REG[040Eh].
When this bit = 0b, PLL1 is disabled and in a power down state. (default)
When this bit = 1b, PLL1 is enabled.

REG[0412h] is Reserved

This register is Reserved and should not be written.

REG[0414h] PLL2 Configuration Register 0
Default = 0000h Read/Write

PLL2 RS bits 3-0				PLL2 VC bits 3-0			
15	14	13	12	11	10	9	8
Reserved		PLL2 V Divider bits 1-0		PLL2 N Multiplier bits 3-0			
7	6	5	4	3	2	1	0

Note

For example, the value of A333h results in a PLL2 output of 23.8MHz when a 5.95MHz reference clock is input to PLL2.

bits 15-12 PLL2 RS bits [3:0]
These bits are used to configure the Low Pass Filter (LPF) resistance and should be set based on the frequency of the PLL2 reference clock.

Table 10-8 : PLL2 RS Configuration

REG[0414h] bits 15-12	PLL2 Reference Clock Frequency
0000b ~ 0111b	Reserved
1000b	20MHz ≤ fPLL2REFCLK ≤ 150MHz
1001b	Reserved
1010b	5MHz ≤ fPLL2REFCLK ≤ 20MHz
1011b ~ 1111b	Reserved

bits 11-8 PLL2 VC bits [3:0]
These bits set the analog adjustment pins for PLL2 and should be set according to the VCO frequency.

Table 10-9 : PLL2 VC Configuration

REG[0414h] bits 11-8	PLL2 VCO Frequency
0000b	Reserved
0001b	100MHz ≤ fVCO ≤ 120MHz
0010b	120MHz < fVCO ≤ 160MHz
0011b	160MHz < fVCO ≤ 200MHz
0100b	200MHz < fVCO ≤ 240MHz
0101b	240MHz < fVCO ≤ 280MHz
0110b	280MHz < fVCO ≤ 320MHz
0111b	320MHz < fVCO ≤ 360MHz
1000b	360MHz < fVCO ≤ 400MHz
1001b ~ 1111b	Reserved

bits 7-6 Reserved
The default value for these bits is 00b.

bits 5-4 PLL2 V Divider bits [1:0]

These bits are used to configure the VCO frequency which must be set between 100MHz and 400MHz. These bits should be set using the following formula.

fVCO = fPLL2OUT x VV

Where:

fVCO is the frequency of VCO, in MHz
fPLL2OUT is the desired PLL2 output frequency, in MHz (see N Multiplier bits)
VV is the value based on the V Divider bits as follows.

Table 10-10 : VV Value

REG[0414h] bits 5-4	VV Value
00b	Reserved
01b	2
10b	4
11b	8

Note

Normally VV is set to 2. When fPLL2OUT is lower than 50MHz, stabilize VCCO by setting VV to 4 or 8. Also, the PLL2 VC bits (REG[0414h] bits 11-8) must be set according to the resulting fVCO. The frequency of VCO (fVCO) must always be within 100MHz ~ 400MHz.

bits 3-0 N Multiplier bits [3:0]

These bits are used to determine the output frequency of PLL2 according to the following formula.

fPLL2OUT = fPLL2REFCLK x NN

Where:

fPLL2OUT is the desired PLL2 output frequency, in MHz
fPLL2REFCLK is the PLL2 reference clock input frequency, in MHz
NN is the N Multiplier value + 1

REG[0416h] PLL2 Configuration Register 1 Default = 0000h							Read/Write
PLL2 Configuration 1 bits 15-8							
15	14	13	12	11	10	9	8
PLL2 Configuration 1 bits 7-0							
7	6	5	4	3	2	1	0

bits 15-0 PLL2 Configuration 1 bits [15:0]

These bits are used to configure PLL2 and should be set to the recommended value of 0040h.

REG[0418h] PLL2 Control Register
Default = 0000h Read/Write

n/a							
15	14	13	12	11	10	9	8
n/a							PLL2 Enable
7	6	5	4	3	2	1	0

bit 0

PLL2 Enable
This bit controls PLL2. PLL2 must be disabled before changing the PLL2 Configuration registers, REG[0414h] ~ REG[0416h].
When this bit = 0b, PLL2 is disabled and in a power down state. (default)
When this bit = 1b, PLL2 is enabled.

REG[041Ah] through REG[0422h] are Reserved

These registers are Reserved and should not be written.

REG[0424h] PLL1 Reference Clock Divide Select Register
Default = 0000h Read/Write

n/a							
15	14	13	12	11	10	9	8
n/a						PLL1 Reference Clock Divide Select bits 1-0	
7	6	5	4	3	2	1	0

bits 1-0

PLL1 Reference Clock Divide Select bits [1:0]
These bits determine the divide ratio applied to the clock used for the PLL1 reference clock. The PLL1 reference clock is derived from the clock input as selected by the CNF[8:7] pins at the rising edge of RESET#. The resulting clock can be used as the PLL1 reference clock (input to PLL1), or used as the PLL2 reference clock (input to PLL2). For detailed information on these options, refer to Section 9, “Clocks” on page 124.

Table 10-11 : PLL1 Reference Clock Divide Ratio Selection

REG[0424h] bits 1-0	PLL1 Source Divide Ratio
00b	1:1 (Default after Reset)
01b	1:2
10b	1:4
11b	1:8

REG[0426h] PLL1 Control Register 0 Default = 0000h							Read/Write
n/a							
15	14	13	12	11	10	9	8
n/a							PLL1 Reference Clock Select
7	6	5	4	3	2	1	0

bit 0 PLL1 Reference Clock Select
This bit selects whether the PLL1 reference clock goes through the divide controlled by REG[0424h] bits 1-0.
When this bit = 0b, the PLL1 reference clock is the clock selected by the CNF[8:7] pins. (default)
When this bit = 1b, the PLL1 reference clock is the divided clock output specified by REG[0424h] bits 1-0.

REG[0428h] PLL1 Control Register 1 Default = 0000h							Read/Write
n/a							
15	14	13	12	11	10	9	8
n/a							PLL1 Output Enable
7	6	5	4	3	2	1	0

bit 0 PLL1 Output Enable
This bit controls the output of PLL1 which is used as the source for SDRAMCLK and SYSCLK (SYSCLK is automatically 1/2 of SDRAMCLK). For further information, refer to Section 9, “Clocks” on page 124.

Before enabling the PLL1 output, the PLL1 Configuration registers (REG[040Ch] ~ REG[040Eh]) must be configured with appropriate values. For further information, refer to the bit descriptions for REG[040Ch] ~ REG[040Eh].
When this bit = 0b, PLL1 output is disabled. (default)
When this bit = 1b, PLL1 output is enabled.

Note

When the PLL1 output is disabled, the PLL1 Reference Clock can be used as the SDRAMCLK and SYSCLK source.

To enable PLL1, the following sequence should be used.

1. Set the PLL1 for a target frequency, program REG[040Ch] ~ REG[040Eh]
2. Enable PLL1, set REG[0410h] bit 0 = 1b
3. Wait for PLL1 output to stabilize
4. Enable PLL1 output, REG[0428h] bit 0 = 1b

REG[042Ch] Key Clock Control Register
Default = 0000h Read/Write

n/a			Key Clock Enable	Key Clock Divide Select bits 11-8			
15	14	13	12	11	10	9	8
Key Clock Divide Select bits 7-0							
7	6	5	4	3	2	1	0

bit 12 Key Clock Enable
This bit controls the clock used for the Keypad Interface (KEYCLK). The clock is derived from SYSCLK using the Key Clock Divide Select bits, REG[042Ch] bits 11-0. For further information on KEYCLK, refer to Section 9, “Clocks” on page 124.
When this bit = 0b, the keypad interface clock (KEYCLK) is disabled.
When this bit = 1b, the keypad interface clock (KEYCLK) is enabled.

bits 11-0 Key Clock Divide Select bits [11:0]
These bits select the divide ratio used to determine the Keypad Interface clock (KEYCLK). The source clock for the KEYCLK is SYSCLK.For further information on KEYCLK, refer to Section 9, “Clocks” on page 124.

Table 10-12 : Keypad Interface Clock Divide Ratio Selection

REG[042Ch] bits 11-0	Divide Ratio
000h	1:1
001h	1:2
002h	1:3
• • •	• • •
FFDh	1:4094
FFEh	1:4095
FFFh	1:4096

REG[042Eh] PWM Source Clock Control Register
Default = 0000h Read/Write

n/a			PWM Source Clock Enable	PWM Source Clock Divide Select bits 11-8			
15	14	13	12	11	10	9	8
PWM Source Clock Divide Select bits 7-0							
7	6	5	4	3	2	1	0

bit 12 PWM Source Clock Enable
This bit controls the clock used for the PWM Interface (PWMSRCCLK). The clock is derived from SYSCLK using the PWM Clock Divide Select bits, REG[042Eh] bits 11-0. For further information on PWMSRCCLK, refer to Section 9, “Clocks” on page 124.
When this bit = 0b, the PWM interface clock (PWMSRCCLK) is disabled.
When this bit = 1b, the PWM interface clock (PWMSRCCLK) is enabled.

bits 11-0 PWM Source Clock Divide Select bits [11:0]
These bits select the divide ratio used to determine the PWM Interface clock (PWMSRCCLK). The source clock for the PWMSRCCLK is SYSCLK. For further information on PWMSRCCLK, refer to Section 9, “Clocks” on page 124.

Table 10-13 : PWM Source Clock Divide Ratio Selection

REG[042Eh] bits 11-0	Divide Ratio
000h	1:1
001h	1:2
002h	1:3
• • •	• • •
FFDh	1:4094
FFEh	1:4095
FFFh	1:4096

REG[0430h] I2C Clock Control Register
Default = 0000h Read/Write

n/a							
15	14	13	12	11	10	9	8
I2C Clock Divide Select bits 7-0							
7	6	5	4	3	2	1	0

bits 7-0 I2C Clock Divide Select bits [7:0]
These bits select the divide ratio used to determine the I2C Interface clock (I2CCLK). The source clock for the I2CCLK is SYSCLK. For further information on I2CCLK, refer to Section 9, “Clocks” on page 124.

I2C Interface Clock Divide Ratio = REG[0430h] bits 7-0 + 1

Table 10-14 : I2C Interface Clock Divide Ratio Selection

REG[0430h] bits 7-0	Divide Ratio
00h	1:1
01h	1:2
02h	1:3
• • •	• • •
FDh	1:254
FEh	1:255
FFh	1:256

REG[0440h] PLL2 Control Register 0
Default = 0000h Read/Write

n/a							
15	14	13	12	11	10	9	8
n/a						PLL2 Source Select bits 1-0	
7	6	5	4	3	2	1	0

bits 1-0 PLL2 Source Select bits [1:0]
These bits select which clock is used for the PLL2 Source. The PLL2 Source is used to derive the clock used for the panel interface. For further information, see Section 9, “Clocks” on page 124.

Table 10-15 : PLL2 Source Selection

REG[0440h] bits 1-0	PLL2 Source
00b (default)	OSC1 Clock
01b	OSC2 Clock
10b	CLKI3 Pin
11b	BUSCLK Pin

REG[0442h] PLL2 Control Register 1
Default = 0000h Read/Write

n/a							
15	14	13	12	11	10	9	8
n/a						PLL2 Reference Clock Divide Select bits 1-0	
7	6	5	4	3	2	1	0

bits 1-0 PLL2 Reference Clock Divide Select bits [1:0]
These bits determine the divide ratio applied to the PLL2 Reference clock. The PLL2 Reference Clock is derived from the clock input as selected by the PLL2 Source Select bits, REG[0440h] bits 1-0. The resulting clock is used as the PLL2 reference clock (input to PLL2). For detailed information on these options, refer to Section 9, “Clocks” on page 124.

Table 10-16 : PLL2 Reference Clock Divide Ratio Selection

REG[0442h] bits 1-0	PLL2 Reference Clock Divide Ratio
00b (default)	1:1
01b	1:2
10b	1:4
11b	1:8

REG[0444h] PLL2 Control Register 2
Default = 0000h Read/Write

n/a							
15	14	13	12	11	10	9	8
n/a				PLL2 Reference Clock Source Select bits 1-0		n/a	PLL2 Output Enable
7	6	5	4	3	2	1	0

bits 3-2 PLL2 Reference Clock Source Select bits [1:0]
These bits select which clock is the source for the PLL2 Reference clock. For further information, refer to Section 9, “Clocks” on page 124.

Table 10-17 : PLL2 Reference Clock Source Selection

REG[0444h] bits 3-2	PLL2 Reference Clock Source
00b (default)	PLL2 Source
01b	PLL2 Divided Source (see REG[0442h] bits 1-0)
10b	Reserved
11b	Reserved

bit 0

PL2 Output Enable

This bit controls the output of PLL2 which is used to derive LCDDCLK or LCDSCLK. For further information, refer to Section 9, “Clocks” on page 124.

Before enabling the PLL2 output, the PLL2 Configuration registers (REG[0414h] ~ REG[0416h]) must be configured with appropriate values. For further information, refer to the bit descriptions for REG[0414h] ~ REG[0416h].
When this bit = 0b, PLL2 output is disabled. (default)
When this bit = 1b, PLL2 output is enabled.

Note

When the PLL2 output is disabled, the PLL2 Reference Clock can be used as the LCDDCLK or LCDSCLK source.

To enable PLL2, the following sequence should be used.

1. Select which clock will be used for the PLL2 reference clock, REG[0444h] bits 3-2
2. Configure REG[0408h], REG[040Ah], REG[0440h], REG[0442h], if necessary
3. Set the PLL2 for a target frequency, program REG[0414h] ~ REG[0416h]
4. Enable PLL2, set REG[0418h] bit 0 = 1b
5. Wait for PLL2 output to stabilize
6. Enable PLL2 output, REG[0444h] bit 0 = 1b

REG[0446h] LCD Clock Control Register 0 Default = 0000h							Read/Write
n/a							
15	14	13	12	11	10	9	8
n/a			LCDDCLK Divide Select bits 4-0				
7	6	5	4	3	2	1	0

bits 4-0 LCDDCLK Divide Select bits [4:0]
These bits select the divide ratio used to generate the LCD panel clock (LCDDCLK) from either the output of PLL2 or the PLL2 reference clock as selected by REG[0444h] bit 0. These bits must only be changed when the LCD interface is inactive, REG[0830h] bit 2 = 0b.

Table 10-18 : LCDDCLK Divide Selection

REG[0446h] bits 4-0	LCDDCLK Divide Ratio	REG[0446h] bits 4-0	LCDDCLK Divide Ratio
00000b (00h)	Reserved	10000b (10h)	Reserved
00001b (01h)	1:2	10001b (11h)	1:18
00010b (02h)	Reserved	10010b (12h)	Reserved
00011b (03h)	1:4	10011b (13h)	1:20
00100b (04h)	Reserved	10100b (14h)	Reserved
00101b (05h)	1:6	10101b (15h)	1:22
00110b (06h)	Reserved	10110b (16h)	Reserved
00111b (07h)	1:8	10111b (17h)	1:24
01000b (08h)	Reserved	11000b (18h)	Reserved
01001b (09h)	1:10	11001b (19h)	1:26
01010b (0Ah)	Reserved	11010b (1Ah)	Reserved
01011b (0Bh)	1:12	11011b (1Bh)	1:28
01100b (0Ch)	Reserved	11100b (1Ch)	Reserved
01101b (0Dh)	1:14	11101b (1Dh)	1:30
01110b (0Eh)	Reserved	11110b (1Eh)	Reserved
01111b (0Fh)	1:16	11111b (1Fh)	1:32

REG[0448h] LCD Clock Control Register 1
Default = 0000h Read/Write

n/a							
15	14	13	12	11	10	9	8
n/a			LCDSCLK Divide Select bits 4-0				
7	6	5	4	3	2	1	0

bits 4-0 LCDSCLK Divide Select bits [4:0]
These bits select the divide ratio used to generate the serial clock (LCDSCLK) for the LCD serial command interface, from either the output of PLL2 or the PLL2 reference clock as selected by REG[0444h] bit 0.

Table 10-19 : LCDSCLK Divide Selection

REG[0448h] bits 4-0	LCDSCLK Divide Ratio	REG[0448h] bits 4-0	LCDSCLK Divide Ratio
00000b (00h)	Reserved	10000b (10h)	Reserved
00001b (01h)	1:2	10001b (11h)	1:18
00010b (02h)	Reserved	10010b (12h)	Reserved
00011b (03h)	1:4	10011b (13h)	1:20
00100b (04h)	Reserved	10100b (14h)	Reserved
00101b (05h)	1:6	10101b (15h)	1:22
00110b (06h)	Reserved	10110b (16h)	Reserved
00111b (07h)	1:8	10111b (17h)	1:24
01000b (08h)	Reserved	11000b (18h)	Reserved
01001b (09h)	1:10	11001b (19h)	1:26
01010b (0Ah)	Reserved	11010b (1Ah)	Reserved
01011b (0Bh)	1:12	11011b (1Bh)	1:28
01100b (0Ch)	Reserved	11100b (1Ch)	Reserved
01101b (0Dh)	1:14	11101b (1Dh)	1:30
01110b (0Eh)	Reserved	11110b (1Eh)	Reserved
01111b (0Fh)	1:16	11111b (1Fh)	1:32

REG[0460h] Software Reset Register
Default = 0000h Read/Write

Software Reset bits 15-8							
15	14	13	12	11	10	9	8
Software Reset bits 7-0							
7	6	5	4	3	2	1	0

bits 15-0 Software Reset bits [15:0]
These bits are used to perform a software reset of the S1D13513. Writing a value of "A55Ah" to these bits causes all synchronous registers to be reset to their default values. A software reset via this register does not modify the contents of any external SDRAM memory.

Note
All data written to this register can be read back, except for A55Ah.

REG[0462h] Clock Enable Register
Default = 0000h Read/Write

n/a							
15	14	13	12	11	10	9	8
n/a		Command FIFO CLK Enable	Reserved	PCLK Enable	HCLK2 Enable	HCLK1 Enable	Reserved
7	6	5	4	3	2	1	0

bit 5 Command FIFO CLK Enable
This bit controls the internal clock used for read/write access to the Command FIFO.
When this bit = 0b, the Command FIFO CLK is disabled.
When this bit = 1b, the Command FIFO CLK is enabled.

bit 4 Reserved
The default value for this bit is 0b..

bit 3 PCLK Enable
This bit controls the internal clock used for read/write access to the synchronous registers. This bit must be enabled once the PLL becomes stable and before accessing any of the synchronous registers (see Section 10.1, "Register Mapping" on page 126). If no synchronous registers will be used, this bit can be disabled to achieve additional power savings.
When this bit = 0b, PCLK is disabled.
When this bit = 1b, PCLK is enabled.

bit 2 HCLK2 Enable
This bit controls the internal clock used for the LCDC and SDRAM. This bit must be enabled once the PLL becomes stable and before using either the LCDC or SDRAM. If **neither** of these will be used, this bit can be disabled to achieve additional power savings.
When this bit = 0b, HCLK2 is disabled.
When this bit = 1b, HCLK2 is enabled.

bit 1 HCLK1 Enable
This bit controls the internal clock used for YUV, YRC, Host, Camera, and SDRAM access. This bit must be enabled once the PLL becomes stable and before accessing or configuring any of these interfaces. If **none** of these interfaces will be used, this bit can be disabled to achieve additional power savings.
When this bit = 0b, HCLK1 is disabled.
When this bit = 1b, HCLK1 is enabled.

bit 0 Reserved
The default value for this bit is 0b.

REG[0464h] GPIOC&D Pull-down Resistor Control Register
Default = 0000h Read/Write

n/a				GPIOD[3:0] Pull-down Control bits 3-0			
15	14	13	12	11	10	9	8
GPIOC[7:0] Pull-down Control bits 7-0							
7	6	5	4	3	2	1	0

bits 11-8 GPIOD[3:0] Pull-down Control bits [3:0]
These bits control the pull-down resistor for each corresponding GPIOD[3:0] pin. For information on the possible usages of the GPIOD[3:0] pins, refer to Section 5.2.4, “GPIO / Multi Function Interface” on page 33 and Section 5.6, “GPIO Pin Mapping” on page 50.
When this bit = 0b, the corresponding pull-down resistor is enabled. (default after reset)
When this bit = 1b, the corresponding pull-down resistor is disabled.

bits 7-0 GPIOC[7:0] Pull-down Control bits [7:0]
These bits control the pull-down resistor for each corresponding GPIOC[7:0] pin. For information on the possible usages of the GPIOC[7:0] pins, refer to Section 5.2.4, “GPIO / Multi Function Interface” on page 33 and Section 5.6, “GPIO Pin Mapping” on page 50.
When this bit = 0b, the corresponding pull-down resistor is enabled. (default after reset)
When this bit = 1b, the corresponding pull-down resistor is disabled.

REG[0466h] is Reserved

This register is Reserved and should not be written.

REG[0468h] GPIOG&H Pull-down Resistor Control Register
Default = 0000h Read/Write

n/a		GPIOH[5:0] Pull-down Control bits 5-0					
15	14	13	12	11	10	9	8
n/a			GPIOG[4:0] Pull-down Control bits 4-0				
7	6	5	4	3	2	1	0

bits 13-8 GPIOH[5:0] Pull-down Control bits [5:0]
These bits control the pull-down resistor for each corresponding GPIOH[5:0] pin. For information on the possible usages of the GPIOH[5:0] pins, refer to Section 5.2.2, “LCD Interface” on page 29 and Section 5.6, “GPIO Pin Mapping” on page 50.
When this bit = 0b, the corresponding pull-down resistor is enabled. (default after reset)
When this bit = 1b, the corresponding pull-down resistor is disabled.

Note
The GPIOH[5:0] pins are multiplexed on the FPDAT[23:18] pins and are not available when 24-bit panels are used.

bits 4-0 GPIOG[4:0] Pull-down Control bits [4:0]
These bits control the pull-down resistor for each corresponding GPIOG[4:0] pin. For information on the possible usages of the GPIOG[4:0] pins, refer to Section 5.2.2, “LCD Interface” on page 29 and Section 5.6, “GPIO Pin Mapping” on page 50.
When this bit = 0b, the corresponding pull-down resistor is enabled. (default after reset)
When this bit = 1b, the corresponding pull-down resistor is disabled.

Note
The GPIOG[4:0] pins are used by some panel types and may not be available for use as general purpose IO pins.

REG[046Ah] MEMDQ Pull-down Resistor Control Register 0
Default = 0000h Read/Write

MEMDQ[15:8] Pull-down Control bits 15-8							
15	14	13	12	11	10	9	8
MEMDQ[7:0] Pull-down Control bits 7-0							
7	6	5	4	3	2	1	0

REG[046Ch] MEMDQ Pull-down Resistor Control Register 1
Default = 0000h Read/Write

MEMDQ[31:24] Pull-down Control bits 31-24							
15	14	13	12	11	10	9	8
MEMDQ[23:16] Pull-down Control bits 23-16							
7	6	5	4	3	2	1	0

bits 15-0 MEMDQ[31:0] Pull-down Control bits [31:0]
These bits control the pull-down resistor for each corresponding MEMDQ[31:0] pin. For detailed pin information, refer to Section 5.2.3, “SDRAM Interface” on page 32.
When this bit = 0b, the corresponding pull-down resistor is enabled. (default after reset)
When this bit = 1b, the corresponding pull-down resistor is disabled.

REG[046Eh] CNF Pull-down Resistor Control Register
Default = 0000h Read/Write

n/a							CNF8 Pull-down Control bit 8
15	14	13	12	11	10	9	8
CNF[7:0] Pull-down Control bits 7-0							
7	6	5	4	3	2	1	0

bits 8-0 CNF[8:0] Pull-down Control bits [8:0]
These bits control the pull-down resistor for each corresponding CNF[8:0] pin. The CNF[8:0] pins are used for configuration of the S1D13513 (see Section 5.3, “Summary of Configuration Options” on page 41). The configuration information is latched at RESET. Then if necessary, the pull-down resistor can be disabled in order to cut constant current via the pull-down resistor (i.e. when a pull-up resistor is attached on a CNF pin).
When this bit = 0b, the corresponding pull-down resistor is enabled. (default after reset)
When this bit = 1b, the corresponding pull-down resistor is disabled.

REG[0470h] Power Down Mode Control Register
Default = 0001h Read/Write

n/a							
15	14	13	12	11	10	9	8
n/a							Power Save Mode Enable
7	6	5	4	3	2	1	0

bit 0 Power Save Mode Enable
This bit controls the state of the software initiated power save mode. When power save mode is disabled, the S1D13513 is operating normally. When power save mode is enabled, the S1D13513 is operating in a power efficient state. In this state, all IO clocks are disabled, but the Host interface bus clock is still enabled.
When this bit = 0b, power save mode is disabled.
When this bit = 1b, power save mode is enabled.

Note

When Power Save Mode is enabled, synchronous registers (see Section 10.1, “Register Mapping” on page 126) and SDRAM memory must not be accessed.

REG[0472h] Bus Time-out Reset Control Register
Default = 0001h Read/Write

n/a							
15	14	13	12	11	10	9	8
n/a					Bus Time-Out Reset Interrupt Flag (RO)	Bus Time-out Reset Interrupt Disable	Bus Time-out Reset Disable
7	6	5	4	3	2	1	0

bit 2 Bus Time-out Reset Interrupt Flag (Read Only)
This bit indicates whether a bus time-out reset has occurred. This bit is masked by the Bus Time-out Reset Interrupt Enable bit, REG[0472h] bit 1, and is cleared by disabling the Bus Time-out Reset Interrupt Enable, REG[0472h] bit 1 = 1b.
When this bit = 0b, a bus time-out reset has not occurred.
When this bit = 1b, a bus time-out reset has occurred.

bit 1 Bus Time-out Reset Interrupt Disable
This bit controls the bus time-out reset interrupt. The status of the bus time-out reset interrupt is indicated by the Bus Time-out Reset Interrupt Flag bit, REG[0472h] bit 2.
When this bit = 0b, the bus time-out reset interrupt is enabled.
When this bit = 1b, the bus time-out reset interrupt is disabled.

bit 0 Bus Time-out Reset Disable
This bit controls the bus time-out reset function which can reset the S1D13513 after a bus time-out occurs. For further information, see Section 20.2, “Host Bus Time-out Function” on page 436.
When this bit = 0b, the bus time-out reset function is enabled and will release WAIT#, if WAIT# is held for more than 2048-3072 PLL1 reference clocks (as selected by CNF[8:7]), before resetting the S1D13513.
When this bit = 1b, the bus time-out reset function is disabled. (default)

REG[04A0h] through REG[04A2h] are Reserved

These registers are Reserved and should not be written.

10.4.3 LCD Panel Configuration Registers

Note

Some pins used by the LCD panel interface are multiplexed with GPIO function pins. Therefore, before enabling the LCD panel interface, the appropriate GPIO pins must be configured for use by the LCD panel interface. For a summary of GPIO pin usage, see Section 5.6, “GPIO Pin Mapping” on page 50.

REG[0800h] LCD Panel Type Select Register
Default = 0000h Read/Write

Reserved	Passive Panel Select	Passive Panel Type Select bits 2-0			Panel Data Bus Width bits 2-0		
15	14	13	12	11	10	9	8
FPSHIFT Polarity Select	n/a				TFT Panel Type Select bits 2-0		
7	6	5	4	3	2	1	0

bit 15 Reserved
The default value for this bit is 0b.

bit 14 Passive Panel Select
This bit selects which panel type the LCD panel interface is configured for.
When this bit = 0b, TFT panel support is selected.
When this bit = 1b, passive panel support is selected.

Note

This bit should be modified only when the LCD Output is disabled, REG[0830h] bit 0 = 0b.

bits 13-11 Passive Panel Type Select bits [2:0]
When a passive panel is selected (REG[0800h] bit 14 = 1b), these bits select the type of passive panel connected to the LCD panel interface.

Table 10-20 : Passive Panel Type Selection

REG[0800h] bits 13-11	Passive Panel Type
000b	Single Monochrome Format (Notes 1 and 2)
001b	Reserved
010b	Single Color Format Type 2 (Notes 1 and 3)
011b ~ 111b	Reserved

Note

1. When a passive panel is selected (REG[0800h] bit 14 = 1b), the clock output on FPSHIFT is equal to:
   LCDDCLK ÷ 8 for Single Monochrome Format STN (REG[0800h] bits 13-11 = 000b)
   LCDDCLK ÷ 4 for Single Color Format 2 STN 8-bit (REG[0800h] bits 13-11 = 010b)
2. In order to display all gray shades when single monochrome format is selected, Pseudo Color Mode must be set to a valid mode (REG[0844h] = 01h, 02h, or 04h) and REG[083Eh] bit 2 must be set to 1b. The maximum number of displayed gray shades for 8 bpp, 16 bpp, or 32 bpp is 64.
3. In order to display all colors when single color format type 2 is selected, Pseudo Color Mode must be set to a valid mode (REG[0844h] = 01h, 02h, or 04h). The maximum number of colors for 32 bpp is 262,144.

bits 10-8 Panel Data Bus Width bits [2:0]
These bits are used to select the data bus width of the selected panel.

Table 10-21 : Panel Data Bus Width Selection

REG[0800h] bits 10-8	Panel Data Bus Width	
	TFT Panel	**Passive Panel**
000b	Reserved	Reserved
001b	16-bit	8-bit
010b	18-bit	Reserved
011b	24-bit	Reserved
100b - 111b	Reserved	Reserved

bit 7 FPSHIFT Polarity Select

This bit selects the polarity of the shift clock for RGB type panels (inverts FPSHIFT). This bit has a different effect based on the setting of the Passive Panel Select bit (REG[0800h] bit 14). Note that changes to the Passive Panel Select bit should be made only when LCD output is disabled, REG[0830h] bit 0 = 0b. If changes are made to the Passive Panel Select bit while LCD output is enabled, it can cause the FPSHIFT signal to become out of phase with the rest of the LCD signals.

For TFT Panels or YUV Digital Out (REG[0800h] bit 14 = 0b):
When this bit = 0b, all panel interface signals change at the falling edge of FPSHIFT for TFT Panels or YUVCLKO (GPIOD2) for YUV Digital Out.
When this bit = 1b, all panel interface signals change at the rising edge of FPSHIFT for TFT Panels or YUVCLKO (GPIOD2) for YUV Digital Out.

For Passive Panels (REG[0800h] bit 14 = 1b):
When this bit = 0b, all panel interface signals change at the rising edge of FPSHIFT.
When this bit = 1b, all panel interface signals change at the falling edge of FPSHIFT.

Note

The polarity of the shift clock (FPSHIFT) cannot be changed while LCD Output is active (REG[0830h] bit 1 = 1b). If a change in polarity is required, use the following steps.

1. Disable LCD output (REG[0830h] bit 0 = 0b)
2. Wait until the LCD interface is no longer active (REG[0830h] bit 2 = 0b)
3. Set the desired polarity of the shift clock (REG[0800h] bit 7)
4. Enable LCD output again (REG[0830h] bit 0 = 1b)

bits 2-0 TFT Panel Type Select bits [2:0]

When a TFT panel is selected (REG[0800h] bit 14 = 0b), these bits select the type of TFT panel connected to the LCD panel interface. For TFT panel types using a Serial Command Interface and HR-TFT panels, the GPIOG[4:0] pins must be configured for the appropriate function using REG[0C1Ah]. For a summary of the functions, see Section 5.6, “GPIO Pin Mapping” on page 50.

Table 10-22 : TFT Panel Type Selection

REG[0800h] bits 2-0	TFT Panel Type Select
000b	General TFT/ND-TFD
001b	Reserved
010b	HR-TFT
011b	Reserved
100b	YUV Digital Output for external video encoder (i.e. ADV-7170) (See Note)
101b - 111b	Reserved

Note

If YUV Digital Output is selected, the bpp mode for any enabled windows (Main, PIP1, PIP2) must be set for YUV Digital Output (see REG[0832h] bits 10-8, 6-4, 2-0). The memory for the windows must also be written as YUV 4:2:2. For further information on YUV Digital Output, refer to Section 21.4, “YUV Digital Output” on page 461.

REG[0802h] LCD Horizontal Total Register
Default = 0000h Read/Write

n/a				Horizontal Total bits 11-8			
15	14	13	12	11	10	9	8
Horizontal Total bits 7-0							
7	6	5	4	3	2	1	0

bits 11-0

Horizontal Total bits [11:0]
These bits specify the Horizontal Total (FPLINE) period, in pixel clock periods. The Horizontal Total is the sum of the Horizontal Display Period and the Horizontal Non-Display Period.

REG[0802h] bits 11-0 = Horizontal Total Period - 1

For monochrome single passive panels
REG[0802h] bits 11-0 = (Horizontal FPSHIFT Clock Number x 8) - 1
For 8-bit color format 2 single passive panels
REG[0802h] bits 11-0 = (Horizontal FPSHIFT Clock Number x 4) -1

Note

This register must be programmed such that the following formula is valid.
HT ≥ HDP + HNDP

REG[0804h] LCD Horizontal Display Period Register
Default = 0000h Read/Write

n/a					Horizontal Display Period bits 10-8		
15	14	13	12	11	10	9	8
Horizontal Display Period bits 7-0							
7	6	5	4	3	2	1	0

bits 10-0

Horizontal Display Period bits [10:0]
These bits specify the Horizontal Display Period, in 2 pixel resolution. The Horizontal Display Period must be less than the Horizontal Total to allow for sufficient Horizontal Non-Display Period.

REG[0804h] bits 10-0 = (Horizontal Display Period ÷ 2) - 1

Note

1. This register must be programmed such that the following formula is valid.
   HT ≥ HDP + HNDP
2. For passive panels (REG[0800h] bit 14 = 1b), the horizontal display period must be a multiple of 8 for monochrome single passive panels and 4 for 8-bit color format 2 single passive panels.

REG[0806h] LCD Horizontal Display Period Start Position Register
Default = 0000h Read/Write

n/a				Horizontal Display Period Start Position bits 11-8			
15	14	13	12	11	10	9	8
Horizontal Display Period Start Position bits 7-0							
7	6	5	4	3	2	1	0

bits 11-0 Horizontal Display Period Start Position bits [11:0]
These bits specify the Horizontal Display Period Start Position, in pixel clock periods.

For TFT:
REG[0806h] bits 11-0 = Horizontal Display Period Start Position - 1
For monochrome single passive panels:
REG[0806h] bits 11-0 = (Horizontal Display Period Start Position x 8) - 1
For 8-bit color format 2 single passive panels:
REG[0806h] bits 11-0 = Horizontal Display Period Start Position x 4) - 1
For YUV Digital Output:
REG[0806h] bits 11-0 = Horizontal Display Period Start Position x 2) - 1

REG[0808h] LCD Horizontal Pulse Width Register
Default = 0000h Read/Write

Horizontal Polarity Select	n/a						Horizontal Pulse Width bit 8
15	14	13	12	11	10	9	8
Horizontal Pulse Width bits 7-0							
7	6	5	4	3	2	1	0

bit 15 Horizontal Polarity Select
This bit selects the polarity of the horizontal sync signal (FPLINE).
When this bit = 0b, the horizontal sync signal (FPLINE) is active low. (default)
When this bit = 1b, the horizontal sync signal (FPLINE) is active high.

bits 8-0 Horizontal Pulse Width bits [8:0]
These bits specify the pulse width of the horizontal sync signal (FPLINE), in pixel clock periods.
REG[0808h] bits 8-0 = Horizontal Pulse Width - 1

For monochrome single passive panels:
REG[0808h] bits 8-0 = (Horizontal Pulse Width x 8) - 1
For 8-bit color format 2 single passive panels:
REG[0808h] bits 8-0 = (Horizontal Pulse Width x 4) - 1

REG[080Ah] LCD Horizontal Pulse Start Position Register							
Default = 0000h							Read/Write
n/a				Horizontal Pulse Start Position bits 11-8			
15	14	13	12	11	10	9	8
Horizontal Pulse Start Position bits 7-0							
7	6	5	4	3	2	1	0

bits 11-0 Horizontal Pulse Start Position bits [11:0]
These bits specify the start position of the horizontal sync pulse (FPLINE), in pixel clock period.
REG[080Ah] bits 11-0 = Horizontal Pulse Start Position

For monochrome single passive panels:
REG[080Ah] bits 11-0 = Horizontal Pulse Start Position x 8
For 8-bit color format 2 single passive panels:
REG[080Ah] bits 11-0 = Horizontal Pulse Start Position x 4

REG[080Ch] LCD Vertical Total Register							
Default = 0000h							Read/Write
n/a				Vertical Total bits 11-8			
15	14	13	12	11	10	9	8
Vertical Total bits 7-0							
7	6	5	4	3	2	1	0

bits 11-0 Vertical Total bits [11:0]
These bits specify the Vertical Total (FPFRAME) period, in lines. The Vertical Total is the sum of the Vertical Display Period and the Vertical Non-Display Period.
REG[080Ch] bits 11-0 = Vertical Total Period in lines - 1

Note
These bits must be set to a value greater than 000h.

REG[080Eh] LCD Vertical Display Period Register							
Default = 0000h							Read/Write
n/a				Vertical Display Period bits 11-8			
15	14	13	12	11	10	9	8
Vertical Display Period bits 7-0							
7	6	5	4	3	2	1	0

bits 11-0 Vertical Display Period bits [11:0]
These bits specify the Vertical Display Period, in lines. The Vertical Display Period must be less than the Vertical Total to allow for sufficient Vertical Non-Display Period.
REG[080Eh] bits 11-0 = Vertical Display Period in lines - 1

Note
These bits must be set to a value less than FFFh.

REG[0810h] LCD Vertical Display Period Start Position Register
Default = 0000h Read/Write

n/a				Vertical Display Period Start Position bits 11-8			
15	14	13	12	11	10	9	8
Vertical Display Period Start Position bits 7-0							
7	6	5	4	3	2	1	0

bits 11-0 Vertical Display Period Start Position bits [11:0]
These bits specify the Vertical Display Period Start Position, in lines.
REG[0810h] bits 11-0 = Vertical Display Period Start Position in lines

REG[0812h] LCD Vertical Pulse Width Register
Default = 0000h Read/Write

Vertical Polarity Select	n/a						
15	14	13	12	11	10	9	8
n/a			Vertical Pulse Width bits 4-0				
7	6	5	4	3	2	1	0

bit 15 Vertical Polarity Select
This bit selects the polarity of the vertical sync signal (FPFRAME).
When this bit = 0b, the vertical sync signal (FPFRAME) is active low. (default)
When this bit = 1b, the vertical sync signal (FPFRAME) is active high.

bits 4-0 Vertical Pulse Width bits [4:0]
These bits specify the pulse width of the vertical sync signal (FPFRAME), in lines.
REG[0812h] bits 4-0 = Vertical Pulse Width in lines - 1

REG[0814h] Vertical Pulse Start Position Register
Default = 0000h Read/Write

n/a				Vertical Pulse Start Position bits 11-8			
15	14	13	12	11	10	9	8
Vertical Pulse Start Position bits 7-0							
7	6	5	4	3	2	1	0

bits 11-0 Vertical Pulse Start Position bits [11:0]
These bits specify the start position of the vertical sync pulse (FPFRAME), in lines.
REG [8014h] bits 11-0 = Vertical Pulse Start Position in lines

REG[0816h] LCD Serial Interface Configuration Register
Default = 0000h Read/Write

Reserved	n/a						
15	14	13	12	11	10	9	8
LCD Serial Command Type bits 2-0			LCD Serial Command Direction	n/a		LCD Serial Clock Phase	LCD Serial Clock Polarity
7	6	5	4	3	2	1	0

bit 15 Reserved
The default value for this bit is 0b.

bits 7-5 LCD Serial Command Type bits [2:0]
These bits determine the serial command type. For AC timing information, see Section 7.6.3, “ND-TFD 8-Bit Serial Interface Timing” on page 96, Section 7.6.4, “ND-TFD 9-Bit Serial Interface Timing” on page 97, Section 7.6.5, “a-Si TFT Serial Interface Timing” on page 98, and Section 7.6.6, “uWIRE Serial Interface Timing” on page 99.

Table 10-23 : LCD Serial Command Type Selection

REG[0816h] bits 7-5	LCD Serial Command Type
000b	ND-TFT 4 pin Serial (8-bit serial data)
001b	ND-TFD 3 pin Serial (9-bit serial data)
010b	a-Si TFT Serial (8-bit serial data)
011b	Reserved
100b	µWire serial (16-bit serial data)
101b	24-bit serial data
110b ~ 111b	Reserved

bit 4 LCD Serial Command Direction
This bit determines the serial command bit direction.
When this bit = 0b, the MSB (most significant bit) is first. (default)
When this bit = 1b, the LSB (least significant bit) is first.

Note

For details on timing, see Section 7.6, “Panel Interface Timing” on page 90 and refer to the appropriate serial interface.

bit 1 LCD Serial Clock Phase
This bit specifies the serial clock phase. For a summary of the serial clock phase and polarity settings, see Table 10-24, “Serial Clock Phase and Polarity,” on page 174.

Note
For details on timing, see Section 7.6, “Panel Interface Timing” on page 90 and refer to the appropriate serial interface.

bit 0 LCD Serial Clock Polarity
This bit specifies the serial clock polarity. For a summary of the serial clock phase and polarity settings, see Table 10-24, “Serial Clock Phase and Polarity,” on page 174.

Table 10-24 : Serial Clock Phase and Polarity

REG[0816h] bit 1	REG[0816h] bit 0	Valid Data	Clock Idling Status
0b	0b	Rising edge of Serial Clock	Low
	1b	Falling edge of Serial Clock	High
1b	0b	Falling edge of Serial Clock	Low
	1b	Rising edge of Serial Clock	High

Note
For details on timing, see Section 7.6, “Panel Interface Timing” on page 90 and refer to the appropriate serial interface.

REG[0818h] LCD Status Register
Default = 0000h Read/Write

VSYNC Interrupt Enable	n/a			VSYNC Interrupt Status	VSYNC Interrupt Mask Disable	n/a	
15	14	13	12	11	10	9	8
n/a							VNDP Status (RO)
7	6	5	4	3	2	1	0

bit 15 VSYNC Interrupt Enable
This bit controls whether the VSYNC Interrupt is output to the Host CPU interface.
When this bit = 0b, the interrupt status is not output to the Host CPU.
When this bit = 1b, the interrupt status is output to the Host CPU.

bit 11 VSYNC Interrupt Status
This bit indicates the status of the VSYNC Interrupt. The VSYNC Interrupt can be masked (or disabled) using the VSYNC Interrupt Mask Disable bit, REG[0818h] bit 10.
When this bit = 0b, a VSYNC Interrupt has not occurred.
When this bit = 1b, a VSYNC Interrupt has occurred.

To clear this bit, write a 1b to this bit.

bit 10 VSYNC Interrupt Mask Disable
This bit determines whether the VSYNC Interrupt is masked. The status of the VSYNC Interrupt is indicated by the VSYNC Interrupt Status bit, REG[0818h] bit 11.
When this bit = 0b, the VSYNC interrupt is masked (the interrupt will not be set).
When this bit = 1b, the VSYNC interrupt is not masked (the interrupt will be set).

bit 0 VNDP Status (Read Only)
This bit indicates whether the LCD panel is in a Vertical Display Period or a Vertical Non-Display Period. To use this bit, the configured VNDP must be greater than 1 line.
When this bit = 0b, the LCD panel output is in a Vertical Display Period.
When this bit = 1b, the LCD panel output is in a Vertical Non-Display Period.

Note
This bit indicates the period between the last horizontal line of data output and 1 line before data output starts again.

REG[081Ah] LCD VSYNC Interrupt Delay Register Default = 0000h							Read/Write
n/a				LCD VSYNC Interrupt Delay bits 11-8			
15	14	13	12	11	10	9	8
LCD VSYNC Interrupt Delay bits 7-0							
7	6	5	4	3	2	1	0

bits 11-0 LCD VSYNC Interrupt Delay bits [11:0]
These bits specify the VSYNC interrupt assertion timing delay from the start of the frame, in lines.

These bits specify the delay, in lines, for the VSYNC interrupt from the start of FPFRAME. The delay can be used by software to prevent display tearing by ensuring that display updates are not overlapping the current line being displayed.

Note
For YUV Digital Output (REG[0800h] bits 2-0 = 100b), the interrupt asserts only for even frames.

REG[081Ch] LCD Serial Command/Parameter Register Default = 0000h							Read/Write
LCD Serial Command bits 7-0							
15	14	13	12	11	10	9	8
LCD Serial Parameter bits 7-0							
7	6	5	4	3	2	1	0

bits 15-8 LCD Serial Command bits [7:0]
These bits are used for 24-bit serial interface and uWire serial interface panels only. When either of these interfaces are enabled (see REG[0816h] bits 7-5), these bits are used as bits 7-0 (CD[15:8]) of the command to the LCD serial interface. For the other panels, bit 8 is only used to determine whether the LCD Serial Parameter bits (REG[081Ch] bits 7-0) contain a command or data.

Note
The serial command is issued to the panel module once REG[081Ch] is written.

bits 7-0 LCD Serial Parameter bits [7:0]
These bits specify the parameter to the LCD serial interface. When 24-bit serial interface mode is enabled (REG[0816h] bits 7-5 = 101b), these bits are used as bits 7-0 (CD[7:0]) of the command to the LCD serial interface.

REG[081Eh] MOD/Serial Command Register							
Default = 0000h							Read/Write
LCD Serial Command bits 15-8							
15	14	13	12	11	10	9	8
MOD Rate bits 7-0							
7	6	5	4	3	2	1	0

bits 15-8 LCD Serial Command bits [15:8]
These bits are used for 24-bit serial interface panels only. When 24-bit serial interface mode is enabled (REG[0816h] bits 7-5 = 101b), these bits are used as bits 15-8 (CD[23:16]) of the command to the LCD serial interface. The serial command is issued to the panel module once REG[081Ch] is written. Therefore, these bits must be set before writing to REG[081Ch].

bits 7-0 MOD Rate bits [7:0]
These bits are used for passive LCD panels only (REG[0800h] bit 14 = 1b).
When these bits equal 00h, the MOD output signal (FPDRDY) toggles every FPFRAME. For a non-zero value *n*, the MOD output signal (FPDRDY) toggles every *n* FPLINE.

10.4.4 HR-TFT Configuration Registers

Note

Some pins used by the LCD panel interface are multiplexed with GPIO function pins. Therefore, before enabling the LCD panel interface, the appropriate GPIO pins must be configured for use by the LCD panel interface. For a summary of GPIO pin usage, see Section 5.6, “GPIO Pin Mapping” on page 50.

REG[0820h] HR-TFT Configuration Register Default = 0000h							Read/Write
n/a							
15	14	13	12	11	10	9	8
n/a					Reserved	HR-TFT PS Mode	HR-TFT Mirror Select
7	6	5	4	3	2	1	0

bit 2 Reserved
The default value for this bit is 0b.

bit 1 HR-TFT PS Mode
These bits are for HR-TFT panels only (REG[0800h] bits 2-0 = 010b) and have no effect for any other panel type. This bit select the timing used for the PS signal. The alternate PS timings (PS1, PS2, PS3) result in additional power savings for the HR-TFT panel.
When this bit = 0b, the PS signal uses PS1 timing.
When this bit = 1b, the PS signal uses PS2 timing.

bit 0 HR-TFT Mirror Select
This bit controls the HR-TFT mirror function.
When this bit = 0b, a normal image is displayed (SPL is used and SPR=0).
When this bit = 1b, a mirror image is displayed (SPL=0 and SPR is used).

REG[0822h] HR-TFT CLS Width Register Default = 0000h							Read/Write
n/a					HR-TFT CLS Pulse Width bits 10-8		
15	14	13	12	11	10	9	8
HR-TFT CLS Pulse Width bits 7-0							
7	6	5	4	3	2	1	0

bits 10-0 HR-TFT CLS Pulse Width bits [10:0]
These bits are for HR-TFT panels only (REG[0800h] bits 2-0 = 010b) and have no effect for any other panel type. These bits determine the width of the CLS signal, in PCLKs.

REG[0824] HR-TFT PS1 Rising Edge Register
Default = 0000h Read/Write

n/a							
15	14	13	12	11	10	9	8
HR-TFT PS1 Rising Edge bits 7-0							
7	6	5	4	3	2	1	0

bits 7-0 PS1 Rising Edge bits [7:0]
These bits are for HR-TFT panels only (REG[0800h] bits 2-0 = 010b) and have no effect for any other panel type. These bits determine the number of PCLKs between the CLS falling edge and the PS1 rising edge.

REG[0826h] HR-TFT PS2 Rising Edge Register
Default = 0000h Read/Write

n/a						HR-TFT PS2 Rising Edge bits 9-8	
15	14	13	12	11	10	9	8
HR-TFT PS2 Rising Edge bits 7-0							
7	6	5	4	3	2	1	0

bits 9-0 HR-TFT PS2 Rising Edge bits [9:0]
These bits are for HR-TFT panels only (REG[0800h] bits 2-0 = 010b) and have no effect for any other panel type. These bits determine the number of PCLKs between the LP falling edge and the first PS2 rising edge.

REG[0828h] HR-TFT PS2 Toggle Width Register
Default = 0000h Read/Write

n/a							HR-TFT PS2 Toggle Width bit 8
15	14	13	12	11	10	9	8
HR-TFT PS2 Toggle Width bits 7-0							
7	6	5	4	3	2	1	0

bits 8-0 HR-TFT PS2 Toggle Width Register [8:0]
These bits are for HR-TFT panels only (REG[0800h] bits 2-0 = 010b) and have no effect for any other panel type. These bits determine the width of the PS2 signal before toggling, in PCLKs.

REG[082Ah] HR-TFT PS3 Signal Width Register
Default = 0000h Read/Write

n/a							HR-TFT PS3 Signal Width bit 8
15	14	13	12	11	10	9	8
HR-TFT PS3 Signal Width bits 7-0							
7	6	5	4	3	2	1	0

bits 8-0 HR-TFT PS3 Signal Width bits [8:0]
These bits are for HR-TFT panels only (REG[0800h] bits 2-0 = 010b) and have no effect for any other panel type. These bits determine the width of the PS3 signal, in PCLKs.

REG[082Ch] HR-TFT REV Toggle Point Register
Default = 0000h Read/Write

n/a							
15	14	13	12	11	10	9	8
n/a	HR-TFT Rev Toggle bits 6-0						
7	6	5	4	3	2	1	0

bits 6-0 HR-TFT REV Toggle bits [6:0]
These bits are for HR-TFT panels only (REG[0800h] bits 2-0 = 010b) and have no effect for any other panel type. These bits determine the width to toggle the REV signal prior to the LP signal, in PCLKs.

REG[082Eh] HR-TFT PS1/2 End Register
Default = 0000h Read/Write

n/a							
15	14	13	12	11	10	9	8
n/a			HR-TFT PS1/2 End bits 4-0				
7	6	5	4	3	2	1	0

bits 4-0 HR-TFT PS1/2 End bits [4:0]
These bits are for HR-TFT panels only (REG[0800h] bits 2-0 = 010b) and have no effect for any other panel type. These bits determine how far the PS signal continues into the vertical non-display period, in lines.

10.4.5 LCD Display Mode Registers

Note

Some pins used by the LCD panel interface are multiplexed with GPIO function pins. Therefore, before enabling the LCD panel interface, the appropriate GPIO pins must be configured for use by the LCD panel interface. For a summary of GPIO pin usage, see Section 5.6, “GPIO Pin Mapping” on page 50.

REG[0830h] Display Mode Setting Register 0 Default = 0000h							Read/Write
Horizontal Doubling Enable	n/a	Display Buffer Manual Select	Displayed Buffer Status (RO)	n/a	Display Mode Select bits 2-0		
15	14	13	12	11	10	9	8
LCD Software Reset (WO)	Display Blank Enable	Video Invert Enable	Reserved	n/a	LCD/YUV Output Status (RO)	n/a	LCD/YUV Output Enable
7	6	5	4	3	2	1	0

Note

Any changes to bits in this register take effect at the end of the current frame.

bit 15 Horizontal Doubling Enable

This bit controls the horizontal doubling function. This bit is used only when YUV output is specified.
When this bit = 0b, horizontal doubling is disabled.
When this bit = 1b, horizontal doubling is enabled.

bit 13 Display Buffer Manual Select

When double buffering is disabled for a specified window (see REG[0834h] bit 15 and REG[0834h] bits 14-13), this bit allows manual selection of which buffer is displayed. This bit has no effect when double buffering is enabled, REG[0834h] bit 15 = 1b.
When this bit = 0b, the front buffer is displayed.
When this bit = 1b, the back buffer is displayed.

bit 12 Displayed Buffer Status (Read Only)

When double buffering is disabled for a specified window (see REG[0834h] bit 15 and REG[0834h] bits 14-13), this bit indicates which buffer is currently being displayed. When double buffering is enabled, this bit is only valid when the sprite engine is selected as the source, REG[0834h] bits 12-11 = 00b.
When this bit = 0b, the front buffer is displayed.
When this bit = 1b, the back buffer is displayed.

bits 10-8 Display Mode Select [2:0]
These bits select the display mode. To avoid any visual tearing, Display Mode settings should only be changed during non-display period which is indicated by REG[0818h] bit 0 or the interrupt status (see REG[0818h] bit 11). For further details, refer to the register descriptions for REG[0818h] bit 15, REG[0818h] bit 10, and REG[081Ah].

Table 10-25 : Display Mode Selection

REG[0830h] bits 10-8	Display Mode
000b	Main Window Only
001b	Main Window + PIP1 Window (see Note)
010b	Main Window + PIP2 Window
011b	Main Window + PIP1 + PIP2 Window (see Note)
100b - 111b	Reserved

Note

The PIP1 window Y start position (REG[084Eh] bits 10-0), must be set to 0 before disabling the PIP1 window.

bit 7 LCD Software Reset (Write Only)
This bit initiates a software reset of the LCD control module. All LCD pins are reset to their reset states and REG[0800h] ~ REG[0870h] are reset to their default values.
Writing a 0b to this bit has no hardware effect.
Writing a 1b to this bit performs a software reset of the LCD control module.

bit 6 Display Blank Enable
This bit blanks the display by forcing all display data outputs low. All display control signals remain unchanged.
When this bit = 0b, the display is active.
When this bit = 1b, the display is blanked.

bit 5 Video Invert Enable
This bit inverts the display by inverting all display data outputs. All display control signals remain unchanged. This bit has no effect if display blank is enabled, REG[0830h] bit 6 = 1b.
When this bit = 0b, the display data is unchanged (normal).
When this bit = 1b, the display data is inverted.

bit 4 Reserved
The default value for this bit is 0b.

bit 2 LCD/YUV Output Status (Read Only)
This bit indicates whether the S1D13513 is outputting to the LCD or YUV interface.
When this bit = 0b, LCD/YUV output is not active.
When this bit = 1b, LCD/YUV output is active.

bit 0 LCD/YUV Output Enable
This bit controls whether the LCD/YUV controls signals and display data are output on the LCD/YUV interfaces. REG[0800h] bits 2-0 control whether the LCD or YUV interface is selected for display data output.
When this bit = 0b, LCD/YUV output is disabled.
When this bit = 1b, LCD/YUV output is enabled.

REG[0832h] Display Mode Setting Register 1 Default = 0000h							Read/Write
PIP Window Priority Control	n/a			PIP2 Window ARGB Format Select	PIP2 Window Bpp Select bits 2-0		
15	14	13	12	11	10	9	8
n/a	PIP1 Window Bpp Select bits 2-0			n/a	Main Window Bpp Select bits 2-0		
7	6	5	4	3	2	1	0

bit 15 PIP Window Priority Control
This bit determines which PIP window is the top layer.
When this bit = 0b, PIP2 is the top layer. (default)
When this bit = 1b, PIP1 is the top layer.

bit 11 PIP2 Window ARGB Format Select
This bit selects the format that PIP2 window display data is stored in the display buffer.
When this bit = 0b, display data is stored as RGB. (normal)
When this bit = 1b, display data is stored as ARGB.

bits 10-8 PIP2 Window Bpp Select bits [2:0]

These bits select the color depth, in bits-per-pixel, for the PIP2 window. The following table lists the available color depths for the PIP2 window, and how the data is stored in the display buffer. The display data can be stored as either RGB or ARGB based on the setting of the PIP2 Window ARGB Format Select bit, REG[0832h] bit 11.

Table 10-26 : PIP2 Window Bpp Selection

REG[0832h] bits 10-8	Color Depth (bpp)	RGB Format REG[0832h] bit 11 = 0b	ARGB Format REG[0832h] bit 11 = 1b
000b	8 bpp	RGB 3:3:2	Reserved
001b	16 bpp	RGB 5:6:5	ARGB 1:5:5:5
010b	16 bpp	YUV Digital Output (Note 3)	ARGB 4:4:4:4
011b	32 bpp	RGB 8:8:8 Unpacked	Reserved
100b	32 bpp	Reserved	ARGB 8:8:8:8
101b ~ 111b	Reserved		

Note

1. When a single color format type 2 panel is selected (REG[0800h] bits 13-11 = 010b) and Pseudo Color Mode is enabled (REG[0844h] = 01h, 02h, or 04h), the maximum number of displayed colors is 262,144 in 32 bpp RGB or 32 bpp ARGB. In 16 bpp RGB or 16 bpp ARGB, the maximum number of displayed colors is 65536.
2. When a single monochrome panel is selected (REG[0800h] bits 13-11 = 000b) and Pseudo Color Mode is enabled (REG[0844h] = 01h, 02h, or 04h), the maximum number of display gray shades is 64 in 8/16/32 bpp, RGB/ARGB.
3. When YUV Digital Output is selected (REG[0800h] bits 2-0 = 100b), these bits must be set to 010b. For further information on YUV Digital Output, refer to Section 21.4, “YUV Digital Output” on page 461.

bits 6-4 PIP1 Window Bpp Selection bits [2:0]
These bits select the color depth, in bits-per-pixel, for the PIP1 window. The following table lists the available color depths for the PIP1 window, and how the data is stored in the display buffer.

Table 10-27 : PIP1 Window Bpp Selection

REG[0832h] bits 6-4	Color Depth (bpp)	Display Buffer Format
000b	8 bpp	RGB 3:3:2
001b	16 bpp	RGB 5:6:5
010b	YUV Digital Output (Note 3)	
011b	32 bpp	RGB 8:8:8 Unpacked
100b	Reserved	
101b ~ 111b	Reserved	

Note

1. When a single color format type 2 panel is selected (REG[0800h] bits 13-11 = 010b) and Pseudo Color Mode is enabled (REG[0844h] = 01h, 02h, or 04h), the maximum number of displayed colors is 262,144 in 32 bpp RGB. In 16 bpp RGB, the maximum number of displayed colors is 65536.
2. When a single monochrome panel is selected (REG[0800h] bits 13-11 = 000b) and Pseudo Color Mode is enabled (REG[0844h] = 01h, 02h, or 04h), the maximum number of display gray shades is 64 in 8/16/32 bpp RGB.
3. When YUV Digital Output is selected (REG[0800h] bits 2-0 = 100b), these bits must be set to 010b. For further information on YUV Digital Output, refer to Section 21.4, “YUV Digital Output” on page 461.

bits 2-0 Main Window Bpp Select bits [2:0]
These bits select the color depth, in bits-per-pixel, for the Main Window. The following table lists the available color depths for the main window, and how the data is stored in the display buffer.

Table 10-28 : Main Window Bpp Selection

REG[0832h] bits 2-0	Color Depth (bpp)	Display Buffer Format
000b	8 bpp	RGB 3:3:2
001b	16 bpp	RGB 5:6:5
010b	YUV Digital Output (Note 3)	
011b	32 bpp	RGB 8:8:8 Unpacked
100b	Reserved	
101b ~ 111b	Reserved	

Note

1. When a single color format type 2 panel is selected (REG[0800h] bits 13-11 = 010b) and Pseudo Color Mode is enabled (REG[0844h] = 01h, 02h, or 04h), the maximum number of displayed colors is 262,144 in 32 bpp RGB. In 16 bpp RGB, the maximum number of displayed colors is 65536.
2. When a single monochrome panel is selected (REG[0800h] bits 13-11 = 000b) and Pseudo Color Mode is enabled (REG[0844h] = 01h, 02h, or 04h), the maximum number of display gray shades is 64 in 8/16/32 bpp RGB.
3. When YUV Digital Output is selected (REG[0800h] bits 2-0 = 100b), these bits must be set to 010b. For further information on YUV Digital Output, refer to Section 21.4, “YUV Digital Output” on page 461.

REG[0834h] Display Mode Setting Register 2
Default = 0000h Read/Write

Double Buffer Enable	Double Buffer Window Select bits 1-0		Double Buffer Write Source Select bits 1-0		PIP2 Window Mirror Enable	PIP2 Window SwivelView Mode Select bits 1-0	
15	14	13	12	11	10	9	8
n/a					View Port (Main + PIP1) Mirror Enable	View Port (Main + PIP1) SwivelView Select bits 1-0	
7	6	5	4	3	2	1	0

bit 15 Double Buffer Enable
This bit controls the double Buffer function which will “automatically” switch between a front and a back buffer to ensure a stable image on the selected window. This function is designed to handle streaming from a camera or other high speed data sources.

Before enabling double buffer, a data source must be selected using the Double Buffer Write Source Select bits (REG[0834h] bits 12-11) and a destination window must be selected using the Double Buffer Window Select bits (REG[0834h] bits 14-13).
When this bit = 0b, double buffer is disabled.
When this bit = 1b, double buffer is enabled.

bits 14-13 Double Buffer Window Select bits [1:0]
These bits select which window will be double buffered.

Table 10-29 : Double Buffer Window Selection

REG[0834h] bits 14-13	Double Buffered Window
00b	Main Window
01b	PIP1 Window
10b	PIP2 Window
11b	Reserved

bits 12-11 Double Buffer Write Source Select bits [1:0]
These bits select the source of the data to be double buffered.

Table 10-30 : Double Buffer Write Source Selection

REG[0834h] bits 12-11	Double Buffer Source
00b	Sprite Engine
01b	Camera Interface
10b	Reserved
11b	Reserved

Note

When the camera interface is selected as the double buffer data source, the YRC must be set to double buffer write mode, REG[3000h] bit 12 = 1b. The YRC Write Start Address 0 (REG[3002h] ~ REG[3004h]) must equal the selected window front buffer address, and the YRC Write Start Address 1 (REG[3006h] ~ REG[3008h]) must equal the selected window back buffer address.

bit 10 PIP2 Window Mirror Enable
This bit controls the Mirror display function for the PIP2 window. For further information on the mirror display function, see Section 13.2, “Mirror Display” on page 385.
When this bit = 0b, mirror display for the PIP2 window is disabled.
When this bit = 1b, mirror display for the PIP2 window is enabled.

bits 9-8 PIP2 Window SwivelView Mode Select bits [1:0]
These bits select the SwivelView mode of the PIP2 window. SwivelView is a counter-clockwise hardware rotation of the PIP2 window. For further information on the SwivelView function, see Section 13.1, “SwivelView™” on page 382.

Table 10-31 : PIP2 Window SwivelView Mode Selection

REG[0834h] bits 9-8	PIP2 Window Swivel View Mode
00b	Normal (0°)
01b	Reserved
10b	180°
11b	Reserved

bit 2 View Port (Main + PIP1) Mirror Enable
This bit controls the Mirror display function for the View Port (Main + PIP1 window). For further information on the mirror display function, see Section 13.2, “Mirror Display” on page 385.
When this bit = 0b, mirror display for the View Port is disabled.
When this bit = 1b, mirror display for the View Port is enabled.

bits 1-0 View Port (Main + PIP1) SwivelView Mode Select bits [1:0]
These bits select the SwivelView mode of the View Port (Main + PIP1). SwivelView is a counter-clockwise hardware rotation of the View Port. For further information on the SwivelView function, see Section 13.1, “SwivelView™” on page 382.

Table 10-32 : View Port (Main + PIP1) SwivelView Mode Selection

REG[0834h] bits 1-0	View Port (Main + PIP1) SwivelView Mode
00b	Normal (0°)
01b	Reserved
10b	180°
11b	Reserved

REG[0836h] PIP2 Window Alpha Blending Mode Register Default = 0000h							Read/Write
n/a	PIP2 Alpha Map Value Shift Enable	PIP2 Alpha Pixel Swap	PIP2 Window Transparent Enable	n/a	PIP2 Window Alpha Blending Mode bits 1-0		PIP2 Window Alpha Blending Enable
15	14	13	12	11	10	9	8
Constant Alpha Value bits 7-0							
7	6	5	4	3	2	1	0

bit 14 PIP2 Alpha Map Value Shift Enable
This bit shifts the alpha map value by one bit to the right. For example, if ARGB 1:5:5:5 format is selected, the converted 8-bit alpha value is either 0h (0% blend) or 80h (50% blend). When this bit is set to 1b, the new alpha value becomes either 0h or C0h (75% blend) respectively.
When this bit = 0b, the alpha map value is not shifted.
When this bit = 1b, the alpha map value is shifted.

Note
This bit has no effect when the alpha map value is 0h or when constant alpha blending is enabled, REG[0836h] bits 10-9 = 00b.

bit 13 PIP2 Alpha Pixel Swap
This bit controls whether the two pixels are swapped in the alpha blending formula listed in REG[0836h] bits 10-9.
When this bit = 0b, PIP2 alpha pixel swapping is disabled.
When this bit = 1b, PIP2 alpha pixel swapping is enabled and the formula changes as follows.

Output Pixel Color = $(Pm \times \alpha) + (Pp \times (1 - \alpha)$
$\alpha = \alpha m \times \alpha c$

Where:
Pm = colors of a pixel in the Main Window
Pp = colors of a pixel in the PIP2 Window
αm = an alpha value in the alpha map
αc = a constant alpha value specified by REG[0836h] bits 7-0

bit 12 PIP2 Window Transparent Enable
This bit controls PIP2 window transparency.
When this bit = 0b, transparency is disabled.
When this bit = 1b, transparency is enabled.

bits 10-9 PIP2 Window Alpha Blending Mode bits [1:0]
These bits determine the alpha blending mode used for the PIP2 window.

Table 10-33 : PIP2 Window Alpha Blending Mode Selection

REG[0836h] bits 10-9	**PIP2 Window Alpha Blending Mode**
00b	Constant Alpha Blending
01b	Blending with Alpha Map (ARGB Format)
10b	Blending with Constant Value and Alpha Map (ARGB Format)
11b	Reserved

Alpha blending is defined by the following formula,

$\text{Output Pixel Colors} = (Pp \times \alpha) + (Pm \times (1 - \alpha)$

$\alpha = \alpha m \times \alpha c$

Where:

Pm = colors of a pixel in the Main Window
Pp = colors of a pixel in the PIP2 Window
αm = an alpha value in the alpha map
αc = a constant alpha value specified by REG[0836h] bits 7-0

If these bits are 00b, $\alpha=\alpha c$. If these bits are 01b, $\alpha = \alpha m$. The α value is normalized to 1 and ranges from 0 to 1. The αm is quantized by the bit width of the alpha value in the ARGB format pixel data, specified by REG[0832h] bits 10-8. For example, if ARGB 1:5:5:5 format is selected, αm is 0 or 0.5 in decimal. If ARGB 4:4:4:4 is selected, αm ranges from 0 to 15/16 in 1/16 increments. If ARGB 8:8:8:8 format is selected, αm ranges from 0 to 255/256 in 1/256 increments.

When alpha blending with alpha map values, REG[0836h] bits 10-9 = 01b or 10b, then ARGB data format must be enabled, REG[0832h] bit 11 = 1b.

bit 8 PIP2 Window Alpha Blending Enable
This bit controls the alpha blending function for the PIP2 window.
When this bit = 0b, alpha blending is disabled.
When this bit = 1b, alpha blending is enabled.

bits 7-0 Constant Alpha Value bits [7:0]
These bits specify the constant alpha value used when the PIP2 Window alpha blending mode requires a constant alpha value (REG[0836h] bits 10-9 = 00b or 10b).

REG[0838h] PIP2 Window Transparent Key Color Red Register							
Default = 0000h							Read/Write
n/a							
15	14	13	12	11	10	9	8
PIP2 Window Transparent Key Color Red bit 7-0							
7	6	5	4	3	2	1	0

bits 7-0 PIP2 Window Transparent Key Color Red bits [7:0]
These bits only have an effect when the PIP2 Window Transparent Enable bit is set, REG[0836h] bit 12 = 1b. These bits set the red color component of the Transparent Key Color.

Note
To match the color of a pixel in the Main Window with a format using less than 8-bits for the red component (i.e. 8 or 16 bpp), the least significant bits must be padded with 0.

REG[083Ah] PIP2 Window Transparent Key Color Green Register							
Default = 0000h							Read/Write
n/a							
15	14	13	12	11	10	9	8
PIP2 Window Transparent Key Color Green bit 7-0							
7	6	5	4	3	2	1	0

bits 7-0 PIP2 Window Transparent Key Color Green bits [7:0]
These bits only have an effect when the PIP2 Window Transparent Enable bit is set, REG[0836h] bit 12 = 1b. These bits set the green color component of the Transparent Key Color.

Note
To match the color of a pixel in the Main Window with a format using less than 8-bits for the green component (i.e. 8 or 16 bpp), the least significant bits must be padded with 0.

REG[083Ch] PIP2 Window Transparent Key Color Blue Register							
Default = 0000h							Read/Write
n/a							
15	14	13	12	11	10	9	8
PIP2 Window Transparent Key Color Blue bit 7-0							
7	6	5	4	3	2	1	0

bits 7-0 PIP2 Window Transparent Key Color Blue bits [7:0]
These bits only have an effect when the PIP2 Window Transparent Enable bit is set, REG[0836h] bit 12 = 1b. These bits set the blue color component of the Transparent Key Color.

Note
To match the color of a pixel in the Main Window with a format using less than 8-bits for the blue component (i.e. 8 or 16 bpp), the least significant bits must be padded with 0.

REG[083Eh] Gamma Control Register
Default = 0000h Read/Write

n/a	Main Window Gamma LUT Bypass Enable	PIP2 Window Gamma LUT Bypass Enable	PIP1 Window Gamma LUT Bypass Enable	Gamma LUT PIP Window Select bits 1-0		Gamma LUT Write Color Select bits 1-0	
15	14	13	12	11	10	9	8
n/a		Gamma LUT Display Bank Select bits 1-0		Input Data Extra Bit Expansion Enable	Input Data Extra Bit Clip Disable	Gamma LUT Address Auto Increment	Gamma LUT Enable
7	6	5	4	3	2	1	0

bit 14 Main Window Gamma LUT Bypass Enable
This bit controls whether the Gamma LUT, as selected by the Gamma LUT Display Bank Select bits (REG[083Eh] bits 5-4), is used to determine the main window image. When the Gamma LUT is bypassed, gamma correction does not take place.
When this bit = 0b, the Gamma LUT is used (not bypassed).
When this bit = 1b, the Gamma LUT is not used (bypassed).

Note
The Main window must not be Gamma LUT enabled if the window is not selected for gamma display in REG[083Eh] bits 11-10 and bits 5-4.

bit 13 PIP2 Window Gamma LUT Bypass Enable
This bit controls whether the Gamma LUT, as selected by the Gamma LUT Display Bank Select bits (REG[083Eh] bits 5-4), is used to determine the PIP2 window image. When the Gamma LUT is bypassed, gamma correction does not take place.
When this bit = 0b, the Gamma LUT is used (not bypassed).
When this bit = 1b, the Gamma LUT is not used (bypassed).

Note
The PIP2 window must not be Gamma LUT enabled if the window is not selected for gamma display in REG[083Eh] bits 11-10 and bits 5-4.

bit 12 PIP1 Window Gamma LUT Bypass Enable
This bit controls whether the Gamma LUT, as selected by the Gamma LUT Display Bank Select bits (REG[083Eh] bits 5-4), is used to determine the PIP1 window image. When the Gamma LUT is bypassed, gamma correction does not take place.
When this bit = 0b, the Gamma LUT is used (not bypassed).
When this bit = 1b, the Gamma LUT is not used (bypassed).

Note
The PIP1 window must not be Gamma LUT enabled if the window is not selected for gamma display in REG[083Eh] bits 11-10 and bits 5-4.

bits 11-10 Gamma LUT PIP Window Select bits [1:0]
These bits only have an effect when REG[083Eh] bits 5-4 = 10b.
When the Gamma LUT Display Bank Select bits are configured for Bank B to be used by the PIP window, these bits allow selection of which PIP window(s) will be gamma corrected.

Table 10-34 : Gamma LUT PIP Window Selection

REG[083Eh] bits 11-10	Gamma LUT PIP Window
00b	PIP1 window + PIP2 window
01b	PIP1 window
10b	PIP2 window
11b	Reserved

bits 9-8 Gamma LUT Write Color Select bits [1:0]
These bits select which RGB component of the Gamma LUT is written to when data is written to the Gamma LUT Access Data Port, REG[0842h]. The address (or index) of the Gamma LUT written to is selected by the Gamma LUT Access Address Port, REG[0840h].

When option 11b is selected, the same data value is written to each component (R, G, and B) of the Gamma LUT.

Table 10-35 : Gamma LUT Write Color Selection

REG[083Eh] bits 9-8	Gamma LUT Write Color
00b	Red Color LUT Write Enable
01b	Green Color LUT Write Enable
10b	Blue Color LUT Write Enable
11b	All Color LUT Write Enable

bits 5-4 Gamma LUT Display Bank Select bits [1:0]
These bits determine how the Gamma LUT banks (A and B) are used for display and accessed by the Host. A Gamma LUT bank cannot be used for gamma correction of the display window and simultaneously programmed by the Host.

Table 10-36 : Gamma LUT Display Bank Selection

REG[083Eh] bits 5-4	Gamma LUT Display Bank
00b	Bank A is used for gamma correction of the View Port (Main + PIP1 Window). Bank B is enabled for accesses from the Host interface.
01b	Bank B is used for gamma correction of the View Port (Main + PIP1 Window). Bank A is enabled for accesses from the Host interface.
10b	Bank A is used for gamma correction of the Main window Bank B is used for gamma correction of the selected PIP Windows (see REG[083Eh] bits 11-10). In this mode, neither Gamma LUT bank can be accessed by the Host interface.
11b	Both Bank A and Bank B are enabled for accesses from Host interface. When writing to the Gamma LUT using the Gamma LUT Access Data Port (REG[0842h]), the same value will be written to both banks of the Gamma LUT. The Gamma LUT should not be read in this mode. In this mode, neither Gamma LUT bank can be used for gamma correction.

bit 3 Input Data Extra Bit Expansion Enable
When this bit = 0b, the extra data bits input to gamma correction are clipped to 0.
When this bit = 1b, the extra data bits input to gamma correction are filled with the MSB bit in each color. For example, if the display buffer format is RGB 5:6:5 and the panel data bus width is 18-bit (R'G'B' 666), the MSB bit, R[5] or B[5], is put in the extra bit of displayed data bit, R'[6] or B'[6], respectively.

bit 2 Input Data Extra Bit Clip Disable
When this bit = 0b, extra bits in displayed data bits at the point after color conversion and before gamma correction, not read from the display buffer, are clipped. For example, if the display buffer format is RGB 5:6:5 and the panel data bus width is 18-bit (RGB 6:6:6), the extra bits of displayed bits, R'[0],B'[0] are clipped before gamma correction.
When this bit = 1b, all data bits input to the gamma correction module are not clipped and are left as is.

Note

When a passive monochrome panel is selected (REG[0800h] bit 14 = 1b and REG[0800h] bits 13-11 = 000b), this bit must be set to 1b, even if the Gamma LUT is bypassed (or disabled).

bit 1 Gamma LUT Address Auto Increment
This bit determines whether the Gamma LUT Access Address Port (REG[0840h]) must be set before each write to the Gamma LUT Access Data Port (REG[0842h]).
When this bit = 0b, the address of the Gamma LUT must be set before each data write.
When this bit = 1b, only the first address of the Gamma LUT must be set. For each subsequent data write, the Gamma LUT address is incremented automatically.

bit 0 Gamma LUT Enable
This bit controls the Gamma LUT function.
When this bit = 0b, the Gamma LUTs are disabled and gamma correction is not done. The Gamma LUTs are also not accessible from the Host in this mode.
When this bit = 1b, the Gamma LUTs are enabled and operate as configured by the bits in REG[083Eh].

REG[0840h] Gamma LUT Access Address Port Register Default = 0000h							Write Only
n/a							
15	14	13	12	11	10	9	8
Gamma LUT Access Address bits 7-0							
7	6	5	4	3	2	1	0

bits 7-0 Gamma LUT Access Address bits [7:0] (Write Only)
These bits specify the address (or index) of the Gamma LUT that will be accessed using the Gamma LUT Access Data Port, REG[0842h]. If more than one Gamma LUT entry will be programmed, the Gamma LUT Address Auto Increment bit can be set (REG[083Eh] bit 1 = 1b) so that the Gamma LUT Access Address must be programmed with the first address only. When auto increment is enabled, after writing the specified value (see REG[083Eh] bits 9-8) to the Gamma LUT, the access address is automatically incremented and the next entry can be programmed immediately.

Note
When reading from the Gamma LUT, these bits must be programmed before each read of the Gamma LUT Access Data Port, REG[0842h].

REG[0842h] Gamma LUT Access Data Port Register Default = XXXXh							Read/Write
n/a							
15	14	13	12	11	10	9	8
Gamma LUT Access Data bits 7-0							
7	6	5	4	3	2	1	0

bits 7-0 Gamma LUT Access Data bits [7:0]
These bits specify the RGB data value to be written to the Gamma LUT at the address (or index) specified by the Gamma LUT Access Address Port, REG[0840h].

Note
The LUT (look-up table) is in SRAM and has an unknown default value. Before using the Gamma LUT, it must be initialized.

REG[0844h] Pseudo Color Mode Register

Default = 0000h Read/Write

n/a							
15	14	13	12	11	10	9	8
n/a					Pseudo Color Mode bits 2-0		
7	6	5	4	3	2	1	0

bits 2-0 Pseudo Color Mode bits [2:0]
These bits select the pseudo color mode that will be used.

Table 10-37 : Pseudo Color Mode Selection

REG[0844h] bits 2-0	Pseudo Color Mode
000b	Disable (each color component is rounded down to Panel Data Bus Width)
001b	2x2 matrix dither enable
010b	FRM enable
011b	Reserved
100b	Error diffusion
101b ~ 111b	Reserved

Note

1. When a passive panel is selected (REG[0800h] bit 14 = 1b), Pseudo Color Mode operation is enabled and is the same for a setting of 01h, 02h or 04h. Pseudo Color Mode operation is disabled for a setting of 00h.
2. When a passive color panel is selected (REG[0800h] bit 14 = 1b and REG[0800h] bits 13-11 = 010b), Pseudo Color Mode should be enabled (REG[0844h] = 01h, 02h or 04h) in order to display 65,536 colors in 16 bpp or 262,144 color in 32 bpp. If Pseudo Color Mode is disabled (REG[0844h] = 00h), the maximum number of colors for 16 bpp and 32 bpp is 4096.
3. When a passive monochrome panel is selected (REG[0800h] bit 14 = 1b and REG[0800h] bits 13-11 = 000b), Pseudo Color Mode should be enabled (REG[0844h] = 01h, 02h or 04h) to display 64 gray shades in 8 bpp, 16 bpp or 32 bpp. If Pseudo Color Mode is disabled (REG[0844h] = 00h), the maximum number of gray shades for 8 bpp, 16 bpp or 32 bpp is 16.

REG[0846h] Display FIFO1 Threshold Register
Default = 01FFh | Read/Write

Reserved	n/a						Display FIFO1 High Threshold bit 8
15	14	13	12	11	10	9	8
Display FIFO1 High Threshold bits 7-0							
7	6	5	4	3	2	1	0

bit 15 Reserved
The default value for this bit is 0b.

bits 8-0 Display FIFO1 High Threshold bits [8:0]
When the difference between the read and write pointer of Display FIFO1 is less than this value, a memory read request is generated.

REG[0848h] Display FIFO2 Threshold Register
Default = 01FFh | Read/Write

Reserved	n/a						Display FIFO2 High Threshold bit 8
15	14	13	12	11	10	9	8
Display FIFO2 High Threshold bits 7-0							
7	6	5	4	3	2	1	0

bit 15 Reserved
The default value for this bit is 0b.

bits 8-0 Display FIFO2 High Threshold bits [8:0]
When the difference between the read and write pointer of Display FIFO2 is less than this value, a memory read request is generated.

REG[084Ah] PIP1 Window X Start Position Register
Default = 0000h | Read/Write

n/a					PIP1 Window X Start Position bits 10-8		
15	14	13	12	11	10	9	8
PIP1 Window X Start Position bits 7-0							
7	6	5	4	3	2	1	0

bits 10-0 PIP1 Window X Start Position bits [10:0]
These bits determine the X start position of the PIP1 window in relation to the origin of the panel, in 2 pixel resolution. The origin is assumed to be 0,0.

REG[084Ah] bits 10-0 = PIP1 window X start position ÷ 2

Note

1. This register must only be changed during Vertical Non-Display Period, REG[0818h] bit 0 = 1b.
2. PIP coordinates must be set within the panel display area.
3. For passive panels REG[0800h] bit 14 = 1b, the PIP1 window X start position must be a multiple of 8.

REG[084Ch] PIP1 Window X End Position Register Default = 0000h							Read/Write
n/a					PIP1 Window X End Position bits 10-8		
15	14	13	12	11	10	9	8
PIP1 Window X End Position bits 7-0							
7	6	5	4	3	2	1	0

bits 10-0 PIP1 Window X End Position bits [10:0]
These bits determine the X end position of the PIP1 window in relation to the origin of the panel, in 2 pixel resolution. The origin is assumed to be 0,0.

REG[084Ch] bits 10-0 = (PIP1 window X end position - 1) ÷ 2

Note

1. This register must only be changed during Vertical Non-Display Period, REG[0818h] bit 0 = 1b.
2. PIP coordinates must be set within the panel display area.
3. For passive panels REG[0800h] bit 14 = 1b, the PIP1 window X end position must be a multiple of 8.

REG[084Eh] PIP1 Window Y Start Position Register

Default = 0000h Read/Write

n/a					PIP1 Window Y Start Position bits 10-8		
15	14	13	12	11	10	9	8
PIP1 Window Y Start Position bits 7-0							
7	6	5	4	3	2	1	0

bits 10-0 PIP1 Window Y Start Position bits [10:0]
These bits determine the Y start position of the PIP1 window in relation to the origin of the panel, in 2 pixel resolution. The origin is assumed to be 0,0.
REG[084Eh] bits 10-0 = PIP1 window Y start position ÷ 2

Note

1. This register must only be changed during Vertical Non-Display Period, REG[0818h] bit 0 = 1b.
2. The PIP1 window Y start position must be set to 0h before disabling the PIP1 window.

REG[0850h] PIP1 Window Y End Position Register

Default = 0000h Read/Write

n/a					PIP1 Window Y End Position bits 10-8		
15	14	13	12	11	10	9	8
PIP1 Window Y End Position bits 7-0							
7	6	5	4	3	2	1	0

bits 10-0 PIP1 Window Y End Position bits [10:0]
These bits determine the Y end position of the PIP1 window in relation to the origin of the panel, in 2 pixel resolution. The origin is assumed to be 0,0.
REG[0850h] bits 10-0 = (PIP1 window Y end position - 1) ÷ 2

Note

This register must only be changed during Vertical Non-Display Period, REG[0818h] bit 0 = 1b.

REG[0852h] PIP2 Window X Start Position Register

Default = 0000h Read/Write

n/a					PIP2 Window X Start Position bits 10-8		
15	14	13	12	11	10	9	8
PIP2 Window X Start Position bits 7-0							
7	6	5	4	3	2	1	0

bits 10-0 PIP2 Window X Start Position bits [10:0]
These bits determine the X start position of the PIP2 window in relation to the origin of the panel, in 2 pixel resolution. The origin is assumed to be 0,0.
REG[0852h] bits 10-0 = PIP2 window X start position ÷ 2

Note

For passive panels (REG[0800h] bit 14 = 1b), the PIP2 window X start position must be a multiple of 8.

REG[0854h] PIP2 Window X End Position Register Default = 0000h							Read/Write
n/a					PIP2 Window X End Position bits 10-8		
15	14	13	12	11	10	9	8
PIP2 Window X End Position bits 7-0							
7	6	5	4	3	2	1	0

bits 10-0 PIP2 Window X End Position bits [10:0]
These bits determine the X end position of the PIP2 window in relation to the origin of the panel, in 2 pixel resolution. The origin is assumed to be 0,0.
REG[0854h] bits 10-0 = (PIP2 window X end position - 1) ÷ 2

Note
For passive panels (REG[0800h] bit 14 = 1b), the PIP2 window X end position must be a multiple of 8.

REG[0856h] PIP2 Window Y Start Position Register Default = 0000h							Read/Write
n/a					PIP2 Window Y Start Position bits 10-8		
15	14	13	12	11	10	9	8
PIP2 Window Y Start Position bits 7-0							
7	6	5	4	3	2	1	0

bits 10-0 PIP2 Window Y Start Position bits [10:0]
These bits determine the Y start position of the PIP2 window in relation to the origin of the panel, in 2 pixel resolution. The origin is assumed to be 0,0.
REG[0852h] bits 10-0 = PIP2 window Y start position ÷ 2

REG[0858h] PIP2 Window Y End Position Register Default = 0000h							Read/Write
n/a					PIP2 Window Y End Position bits 10-8		
15	14	13	12	11	10	9	8
PIP2 Window Y End Position bits 7-0							
7	6	5	4	3	2	1	0

bits 10-0 PIP2 Window Y End Position bits [10:0]
These bits determine the Y end position of the PIP2 window in relation to the origin of the panel, in 2 pixel resolution. The origin is assumed to be 0,0.
REG[0858h] bits 10-0 = (PIP2 window Y end position - 1) ÷ 2

REG[085Ah] Main Window Front Buffer Start Address Register 0 Default = 0000h							Read/Write
Main Window Front Buffer Start Address bits 15-8							
15	14	13	12	11	10	9	8
Main Window Front Buffer Start Address bits 7-0							
7	6	5	4	3	2	1	0

REG[085Ch] Main Window Front Buffer Start Address Register 1 Default = 0000h							Read/Write
n/a						Main Window Front Buffer Start Address bits 25-24	
15	14	13	12	11	10	9	8
Main Window Front Buffer Start Address bits 23-16							
7	6	5	4	3	2	1	0

REG[085Ch] bits 9-0
REG[085Ah] bits 15-0 Main Window Front Buffer Start Address bits [25:0]
These bits specify the memory start address of the Main Window front buffer, in bytes.

Note
For 32 bpp (REG[0832h] bits 2-0 = 011b), bits 1-0 must be set to 00b.
For 16 bpp (REG[0832h] bits 2-0 = 001b), bit 0 must be set to 0b.

REG[085Eh] PIP1 Window Front Buffer Start Address Register 0 Default = 0000h							Read/Write
PIP1 Window Front Buffer Start Address bits 15-8							
15	14	13	12	11	10	9	8
PIP1 Window Front Buffer Start Address bits 7-0							
7	6	5	4	3	2	1	0

REG[0860h] PIP1 Window Front Buffer Start Address Register 1 Default = 0000h							Read/Write
n/a						PIP1 Window Front Buffer Start Address bits 25-24	
15	14	13	12	11	10	9	8
PIP1 Window Front Buffer Start Address bits 23-16							
7	6	5	4	3	2	1	0

REG[0860h] bits 9-0
REG[085Eh] bits 15-0 PIP1 Window Front Buffer Start Address bits [25:0]
These bits specify the memory start address of the PIP1 Window front buffer, in bytes.

Note
For 32 bpp (REG[0832h] bits 6-4 = 011b), bits 1-0 must be set to 00b.
For 16 bpp (REG[0832h] bits 6-4 = 001b), bit 0 must be set to 0b.

REG[0862h] PIP2 Window Front Buffer Start Address Register 0
Default = 0000h Read/Write

PIP2 Window Front Buffer Start Address bits 15-8							
15	14	13	12	11	10	9	8
PIP2 Window Front Buffer Start Address bits 7-0							
7	6	5	4	3	2	1	0

REG[0864h] PIP2 Window Front Buffer Start Address Register 1
Default = 0000h Read/Write

n/a						PIP2 Window Front Buffer Start Address bits 25-24	
15	14	13	12	11	10	9	8
PIP2 Window Front Buffer Start Address bits 23-16							
7	6	5	4	3	2	1	0

REG[0864h] bits 9-0
REG[0862h] bits 15-0 PIP2 Window Front Buffer Start Address bits [25:0]
These bits specify the memory start address of the PIP2 Window front buffer, in bytes.

Note

For 32 bpp (see REG[0832h] bits 10-8), bits 1-0 must be set to 00b.
For 16 bpp (see REG[0832h] bits 10-8), bit 0 must be set to 0b.

REG[0866h] Main/PIP1/PIP2 Window Back Buffer Start Address Register 0
Default = 0000h Read/Write

Main/PIP1/PIP2 Window Back Buffer Start Address bits 15-8							
15	14	13	12	11	10	9	8
Main/PIP1/PIP2 Window Back Buffer Start Address bits 7-0							
7	6	5	4	3	2	1	0

REG[0868h] Main/PIP1/PIP2 Window Back Buffer Start Address Register 1
Default = 0000h Read/Write

n/a						Main/PIP1/PIP2 Window Back Buffer Start Address bits 25-24	
15	14	13	12	11	10	9	8
Main/PIP1/PIP2 Window Back Buffer Start Address bits 23-16							
7	6	5	4	3	2	1	0

REG[0868h] bits 9-0
REG[0866h] bit 15-0 Main/PIP1/PIP2 Window Back Buffer Start Address bits [25:0]
These bits specify the memory start address of the Main/PIP1/PIP2 back buffer, in bytes.

Note

For 32 bpp (see REG[0832h]), bits 1-0 must be set to 00b.
For 16 bpp (see REG[0832h]), bit 0 must be set to 0b.

REG[086Ah] Main Window Front Buffer Line Address Offset Register
Default = 0000h Read/Write

n/a		Main Window Front Buffer Line Address Offset bits 13-8					
15	14	13	12	11	10	9	8
Main Window Front Buffer Line Address Offset bits 7-0							
7	6	5	4	3	2	1	0

bits 13-0 Main Window Front Buffer Line Address Offset bits [13:0]
These bits specify the offset from the beginning of one display line to the beginning of the next display line in the memory used for the Main Window front buffer, in bytes.

REG[086Ch] PIP1 Window Front Buffer Line Address Offset Register
Default = 0000h Read/Write

n/a		PIP1 Window Front Buffer Line Address Offset bits 13-8					
15	14	13	12	11	10	9	8
PIP1 Window Front Buffer Line Address Offset bits 7-0							
7	6	5	4	3	2	1	0

bits 13-0 PIP1 Window Front Buffer Line Address Offset bits [13:0]
These bits specify the offset from the beginning of one display line to the beginning of the next display line in the memory used for the PIP1 Window front buffer, in bytes.

Note

This register must only be changed during Vertical Non-Display Period, REG[0818h] bit 0 = 1b.

REG[086Eh] PIP2 Window Front Buffer Line Address Offset Register
Default = 0000h Read/Write

n/a		PIP2 Window Front Buffer Line Address Offset bits 13-8					
15	14	13	12	11	10	9	8
PIP2 Window Front Buffer Line Address Offset bits 7-0							
7	6	5	4	3	2	1	0

bits 13-0 PIP2 Window Front Buffer Line Address Offset bits [13:0]
These bits specify the offset from the beginning of one display line to the beginning of the next display line in the memory used for the PIP2 Window front buffer, in bytes.

REG[0870h] Main/PIP1/PIP2 Window Back Buffer Line Address Offset Register
Default = 0000h Read/Write

n/a		Main/PIP1/PIP2 Window Back Buffer Line Address Offset bits 13-8					
15	14	13	12	11	10	9	8
Main/PIP1/PIP2 Window Back Buffer Line Address Offset bits 7-0							
7	6	5	4	3	2	1	0

bits 13-0 Main/PIP1/PIP2 Window Back Buffer Line Address Offset bits [13:0]
These bits specify the offset from the beginning of one display line to the beginning of the next display line in the memory used for the Main/PIP1/PIP2 Window back buffer, in bytes.

REG[0880h] Color Conversion Control Register Default = 0000h							Read/Write
n/a							
15	14	13	12	11	10	9	8
UV Offset Enable	n/a		Color Conversion Window Select bits 1-0		n/a		Color Conversion Enable
7	6	5	4	3	2	1	0

bit 7 UV Offset Enable
When this bit = 0b, UV offset is disabled.
When this bit = 1b, UV offset is enabled.

bits 4-3 Color Conversion Window Select bits [1:0]
When color conversion is enabled (REG[0880h] bit 0 = 1b), these bits select which window color conversion is applied to.

Table 10-38 : Color Conversion Window Selection

REG[0880h] bits 4-3	Color Conversion is Applied to
00b	Main, PIP1, and PIP2 windows
01b	PIP1 window only
10b	PIP2 window only
11b	PIP1 and PIP2 windows only

bit 0 Color Conversion Enable
This bit controls the color conversion function.
When this bit = 0b, color conversion is disabled (or bypassed).
When this bit = 1b, color conversion is enabled.

REG[0882h] ~ REG[0892h] Color Conversion Matrix Coefficient Registers 0-8 Default = 0000h							Read/Write
n/a					Color Conversion Matrix Coefficient X bits 10-8		
15	14	13	12	11	10	9	8
Color Conversion Matrix Coefficient X bits 7-0							
7	6	5	4	3	2	1	0

bits 10-0 Color Conversion Matrix Coefficient X bits [10:0]
These bits form the matrix coefficients used for color conversion. Bit 10 is the sign bit. Bits 9-8 are the integer, and bits 7-0 are the fractional part.

10.4.6 GPIO Registers

Note

All GPIO function pins are multiplexed with other function pins. After reset, the GPIO pins must be configured according to the alternate functions required by the implementation. For detailed pin descriptions, refer to Section 5, “Pins” on page 21. For a summary of GPIO pin usage, see Section 5.6, “GPIO Pin Mapping” on page 50.

REG[0C00h] GPIOA Data Register Default = 0000h							Read/Write
n/a							
15	14	13	12	11	10	9	8
GPIOA Data bits 7-0							
7	6	5	4	3	2	1	0

bits 7-0

GPIOA Data bits [7:0]
These bits are used differently based on the configuration of the corresponding GPIOA[7:0] pin. To determine the configuration of the GPIOAx pin, refer to the GPIOA Pin Function register, REG[0C02h].
For Reads:
When the GPIOAx pin is configured as an input, the corresponding GPIOA Data bit indicates the input state of the pin. For example, if GPIOA7 is high, GPIOA Data bit 7 will return a 1b.
When the GPIOAx pin is configured as an output, the corresponding GPIOA Data bit indicates the value of the register bit.
For Writes:
When the GPIOAx pin is configured as an output, the corresponding GPIOA Data bit controls the output level of the pin. For example, if GPIOA Data bit 7 is set to 1b, GPIOA7 will output high.

REG[0C02h] GPIOA Pin Function Register Default = 0000h							Read/Write
GPIOA7 Mode bits 1-0		GPIOA6 Mode bits 1-0		GPIOA5 Mode bits 1-0		GPIOA4 Mode bits 1-0	
15	14	13	12	11	10	9	8
GPIOA3 Mode bits 1-0		GPIOA2 Mode bits 1-0		GPIOA1 Mode bits 1-0		GPIOA0 Mode bits 1-0	
7	6	5	4	3	2	1	0

bits 15-0 GPIOAx Mode bits [1:0]
Each GPIOA[7:0] pin can be configured as an input, output, or one of two non-GPIO functions. These bits specify the pin function (or mode) for the corresponding GPIOAx pin. For details on the non-GPIO functions assigned to each GPIOAx pin, see Section 5.2.4, “GPIO / Multi Function Interface” on page 33.

Table 10-39 : GPIOAx Pin Function

GPIOAx Mode bits 1-0	GPIOAx Pin Function
00b (default)	GPIOAx is configured as an input
01b	GPIOAx is configured for Non-GPIO Function #1
10b	GPIOAx is configured as an output
11b	GPIOAx is configured for Non-GPIO Function #2

REG[0C04h] GPIOB Data Register
Default = 0000h Read/Write

n/a							
15	14	13	12	11	10	9	8
GPIOB Data bits 7-0							
7	6	5	4	3	2	1	0

bits 7-0 GPIOB Data bits [7:0]
These bits are used differently based on the configuration of the corresponding GPIOB[7:0] pin. To determine the configuration of the GPIOBx pin, refer to the GPIOB Pin Function register, REG[0C06h].
For Reads:
When the GPIOBx pin is configured as an input, the corresponding GPIOB Data bit indicates the input state of the pin. For example, if GPIOB7 is high, GPIOB Data bit 7 will return a 1b.
When the GPIOBx pin is configured as an output, the corresponding GPIOB Data bit indicates the value of the register bit.
For Writes:
When the GPIOBx pin is configured as an output, the corresponding GPIOB Data bit controls the output level of the pin. For example, if GPIOB Data bit 7 is set to 1b, GPIOB7 will output high.

REG[0C06h] GPIOB Pin Function Register
Default = 0000h Read/Write

GPIOB7 Mode bits1-0		GPIOB6 Mode bits 1-0		GPIOB5 Mode bits 1-0		GPIOB4 Mode bits 1-0	
15	14	13	12	11	10	9	8
GPIOB3 Mode bits 1-0		GPIOB2 Mode bits 1-0		GPIOB1 Mode bits 1-0		GPIOB0 Mode bits 1-0	
7	6	5	4	3	2	1	0

bits 15-0 GPIOBx Mode bits [1:0]
Each GPIOB[7:0] pin can be configured as an input, output, or one of two non-GPIO functions. These bits specify the pin function (or mode) for the corresponding GPIOBx pin. For details on the non-GPIO functions assigned to each GPIOBx pin, see Section 5.2.4, “GPIO / Multi Function Interface” on page 33.

Table 10-40 : GPIOBx Pin Function

GPIOBx Mode bits 1-0	GPIOBx Pin Function
00b (default)	GPIOBx is configured as an input
01b	GPIOBx is configured for Non-GPIO Function #1
10b	GPIOBx is configured as an output
11b	GPIOBx is configured for Non-GPIO Function #2

Note
For GPIOB[6:4], the Non-GPIO Function #2 is Reserved and should not be selected.

REG[0C08h] GPIOC Data Register
Default = 0000h Read/Write

n/a							
15	14	13	12	11	10	9	8
GPIOC Data bits 7-0							
7	6	5	4	3	2	1	0

bits 7-0 GPIOC Data bits [7:0]
These bits are used differently based on the configuration of the corresponding GPIOC[7:0] pin. To determine the configuration of the GPIOCx pin, refer to the GPIOC Pin Function register, REG[0C0Ah].
For Reads:
When the GPIOCx pin is configured as an input, the corresponding GPIOC Data bit indicates the input state of the pin. For example, if GPIOC7 is high, GPIOC Data bit 7 will return a 1b.
When the GPIOCx pin is configured as an output, the corresponding GPIOC Data bit indicates the value of the register bit.
For Writes:
When the GPIOCx pin is configured as an output, the corresponding GPIOC Data bit controls the output level of the pin. For example, if GPIOC Data bit 7 is set to 1b, GPIOC7 will output high.

REG[0C0Ah] GPIOC Pin Function Register
Default = 0000h Read/Write

GPIOC7 Mode bits1-0		GPIOC6 Mode bits 1-0		GPIOC5 Mode bits 1-0		GPIOC4 Mode bits 1-0	
15	14	13	12	11	10	9	8
GPIOC3 Mode bits 1-0		GPIOC2 Mode bits 1-0		GPIOC1 Mode bits 1-0		GPIOC0 Mode bits 1-0	
7	6	5	4	3	2	1	0

bits 15-0 GPIOCx Mode bits [1:0]
Each GPIOC[7:0] pin can be configured as an input, output, or one of two non-GPIO functions. These bits specify the pin function (or mode) for the corresponding GPIOCx pin. For details on the non-GPIO functions assigned to each GPIOCx pin, see Section 5.2.4, “GPIO / Multi Function Interface” on page 33.

Table 10-41 : GPIOCx Pin Function

GPIOCx Mode bits 1-0	GPIOCx Pin Function
00b (default)	GPIOCx is configured as an input
01b	GPIOCx is configured for Non-GPIO Function #1
10b	GPIOCx is configured as an output
11b	GPIOCx is configured for Non-GPIO Function #2

REG[0C0Ch] GPIOD Data Register Default = 0000h							Read/Write
n/a							
15	14	13	12	11	10	9	8
n/a				GPIOD Data bits 3-0			
7	6	5	4	3	2	1	0

bits 3-0 GPIOD Data bits [3:0]
These bits are used differently based on the configuration of the corresponding GPIOD[3:0] pin. To determine the configuration of the GPIODx pin, refer to the GPIOD Pin Function register, REG[0C0Eh].
For Reads:
When the GPIODx pin is configured as an input, the corresponding GPIOD Data bit indicates the input state of the pin. For example, if GPIOD3 is high, GPIOD Data bit 3 will return a 1b.
When the GPIODx pin is configured as an output, the corresponding GPIOD Data bit indicates the value of the register bit.
For Writes:
When the GPIODx pin is configured as an output, the corresponding GPIOD Data bit controls the output level of the pin. For example, if GPIOD Data bit 3 is set to 1b, GPIOD3 will output high.

REG[0C0Eh] GPIOD Pin Function Register Default = 0000h							Read/Write
n/a							
15	14	13	12	11	10	9	8
GPIOD3 Mode bits 1-0		GPIOD2 Mode bits 1-0		GPIOD1 Mode bits 1-0		GPIOD0 Mode bits 1-0	
7	6	5	4	3	2	1	0

bits 7-0 GPIODx Mode bits [1:0]
Each GPIOD[3:0] pin can be configured as an input, output, or one of two non-GPIO functions. These bits specify the pin function (or mode) for the corresponding GPIODx pin. For details on the non-GPIO functions assigned to each GPIODx pin, see Section 5.2.4, "GPIO / Multi Function Interface" on page 33.

Table 10-42 : GPIODx Pin Function

GPIODx Mode bits 1-0	GPIODx Pin Function
00b (default)	GPIODx is configured as an input
01b	GPIODx is configured for Non-GPIO Function #1
10b	GPIODx is configured as an output
11b	GPIODx is configured for Non-GPIO Function #2

REG[0C10h] through REG[0C16h] are Reserved

These registers are Reserved and should not be written.

REG[0C18h] GPIOG Data Register
Default = 0000h Read/Write

n/a							
15	14	13	12	11	10	9	8
n/a			GPIOG Data bits 4-0				
7	6	5	4	3	2	1	0

bits 4-0 GPIOG Data bits [4:0]
These bits are used differently based on the configuration of the corresponding GPIOG[4:0] pin. To determine the configuration of the GPIOGx pin, refer to the GPIOG Pin Function register, REG[0C1Ah].
For Reads:
When the GPIOGx pin is configured as an input, the corresponding GPIOG Data bit indicates the input state of the pin. For example, if GPIOG4 is high, GPIOG Data bit 4 will return a 1b.
When the GPIOGx pin is configured as an output, the corresponding GPIOG Data bit indicates the value of the register bit.
For Writes:
When the GPIOGx pin is configured as an output, the corresponding GPIOG Data bit controls the output level of the pin. For example, if GPIOG Data bit 4 is set to 1b, GPIOG4 will output high.

REG[0C1Ah] GPIOG Pin Function Register
Default = 0000h Read/Write

n/a						GPIOG4 Mode bits 1-0	
15	14	13	12	11	10	9	8
GPIOG3 Mode bits 1-0		GPIOG2 Mode bits 1-0		GPIOG1 Mode bits 1-0		GPIOG0 Mode bits 1-0	
7	6	5	4	3	2	1	0

bits 9-0 GPIOGx Mode bits [1:0]
Each GPIOG[4:0] pin can be configured as an input, output, or one of two non-GPIO functions. These bits specify the pin function (or mode) for the corresponding GPIOGx pin. For details on the non-GPIO functions assigned to each GPIOGx pin, see Section 5.2.4, “GPIO / Multi Function Interface” on page 33.

Table 10-43 : GPIOGx Pin Function

GPIOGx Mode bits 1-0	GPIOGx Pin Function
00b (default)	GPIOGx is configured as an input
01b	GPIOGx is configured for Non-GPIO Function #1
10b	GPIOGx is configured as an output
11b	GPIOGx is configured for Non-GPIO Function #2

REG[0C1Ch] GPIOH Data Register
Default = 00XXh Read/Write

n/a							
15	14	13	12	11	10	9	8
n/a		GPIOH Data bits 5-0					
7	6	5	4	3	2	1	0

bits 5-0 GPIOH Data bits [5:0]
These bits are used differently based on the configuration of the corresponding GPIOH[5:0] pin. To determine the configuration of the GPIOHx pin, refer to the GPIOH Pin Function register, REG[0C1Eh].
For Reads:
When the GPIOHx pin is configured as an input, the corresponding GPIOH Data bit indicates the input state of the pin. For example, if GPIOH5 is high, GPIOH Data bit 5 will return a 1b.
When the GPIOHx pin is configured as an output, the corresponding GPIOH Data bit indicates the value of the register bit.
For Writes:
When the GPIOHx pin is configured as an output, the corresponding GPIOH Data bit controls the output level of the pin. For example, if GPIOH Data bit 5 is set to 1b, GPIOH5 will output high.

REG[0C1Eh] GPIOH Pin Function Register
Default = 0000h Read/Write

n/a				GPIOH5 Mode bits 1-0		GPIOH4 Mode bits 1-0	
15	14	13	12	11	10	9	8
GPIOH3 Mode bits 1-0		GPIOH2 Mode bits 1-0		GPIOH1 Mode bits 1-0		GPIOH0 Mode bits 1-0	
7	6	5	4	3	2	1	0

bits 11-0 GPIOHx Mode bits [1:0]
Each GPIOH[5:0] pin can be configured as an input, output, or one of two non-GPIO functions. These bits specify the pin function (or mode) for the corresponding GPIOHx pin. For details on the non-GPIO functions assigned to each GPIOHx pin, see Section 5.2.4, “GPIO / Multi Function Interface” on page 33.

Table 10-44 : GPIOHx Pin Function

GPIOHx Mode bits 1-0	GPIOHx Pin Function
00b (default)	GPIOHx is configured as an input
01b	GPIOHx is configured for Non-GPIO Function #1
10b	GPIOHx is configured as an output
11b	GPIOHx is configured for Non-GPIO Function #2

REG[0C20h] through REG[0C22h] are Reserved

These registers are Reserved and should not be written.

REG[0C24h] GPIOA&B Interrupt Type Register
Default = 0000h Read/Write

GPIOB Interrupt Type Select bits 7-0							
15	14	13	12	11	10	9	8
GPIOA Interrupt Type Select bits 7-0							
7	6	5	4	3	2	1	0

bits 15-8 GPIOB Interrupt Type Select bits [7:0]
These bits individually control the trigger type of the interrupts associated with the GPIOB[7:0] pins. The interrupt function is controlled using REG[0C28h].
When this bit = 0b, the interrupt uses a level trigger.
When this bit = 1b, the interrupt uses an edge trigger.

For example when bit 15 = 1b, the interrupt associated with the GPIOB7 pin uses an edge trigger.

Note

Before changing any of these bits, all outstanding interrupts must be cleared using REG[0C2Ah].

bits 7-0 GPIOA Interrupt Type Select bits [7:0]
These bits individually control the trigger type of the interrupts associated with the GPIOA[7:0] pins. The interrupt function is controlled using REG[0C28h].
When this bit = 0b, the interrupt uses a level trigger.
When this bit = 1b, the interrupt uses a edge trigger.

For example when bit 0 = 0b, the interrupt associated with the GPIOA0 pin uses a level trigger.

Note

Before changing any of these bits, all outstanding interrupts must be cleared using REG[0C2Ah].

REG[0C26h] GPIOA&B Interrupt Polarity Register Default = 0000h							Read/Write
GPIOB Interrupt Polarity Select bits 7-0							
15	14	13	12	11	10	9	8
GPIOA Interrupt Polarity Select bits 7-0							
7	6	5	4	3	2	1	0

bits 15-8 GPIOB Interrupt Polarity Select bits [7:0]
These bits individually control the polarity of the interrupts associated with the GPIOB[7:0] pins. The interrupt function is controlled using REG[0C28h].
When this bit = 0b, the interrupt is triggered when high (for level trigger) or when rising (for edge trigger).
When this bit = 1b, the interrupt is triggered when low (for level trigger) or when falling (for edge trigger).

Note

Before changing any of these bits, all outstanding interrupts must be cleared using REG[0C2Ah].

bits 7-0 GPIOA Interrupt Polarity Select bits [7:0]
These bits individually control the polarity of the interrupts associated with the GPIOA[7:0] pins. The interrupt function is controlled using REG[0C28h].
When this bit = 0b, the interrupt is triggered when high (for level trigger) or when rising (for edge trigger).
When this bit = 1b, the interrupt is triggered when low (for level trigger) or when falling (for edge trigger).

Note

Before changing any of these bits, all outstanding interrupts must be cleared using REG[0C2Ah].

REG[0C28h] GPIOA&B Interrupt Enable Register Default = 0000h							Read/Write
GPIOB Interrupt Enable bits 7-0							
15	14	13	12	11	10	9	8
GPIOA Interrupt Enable bits 7-0							
7	6	5	4	3	2	1	0

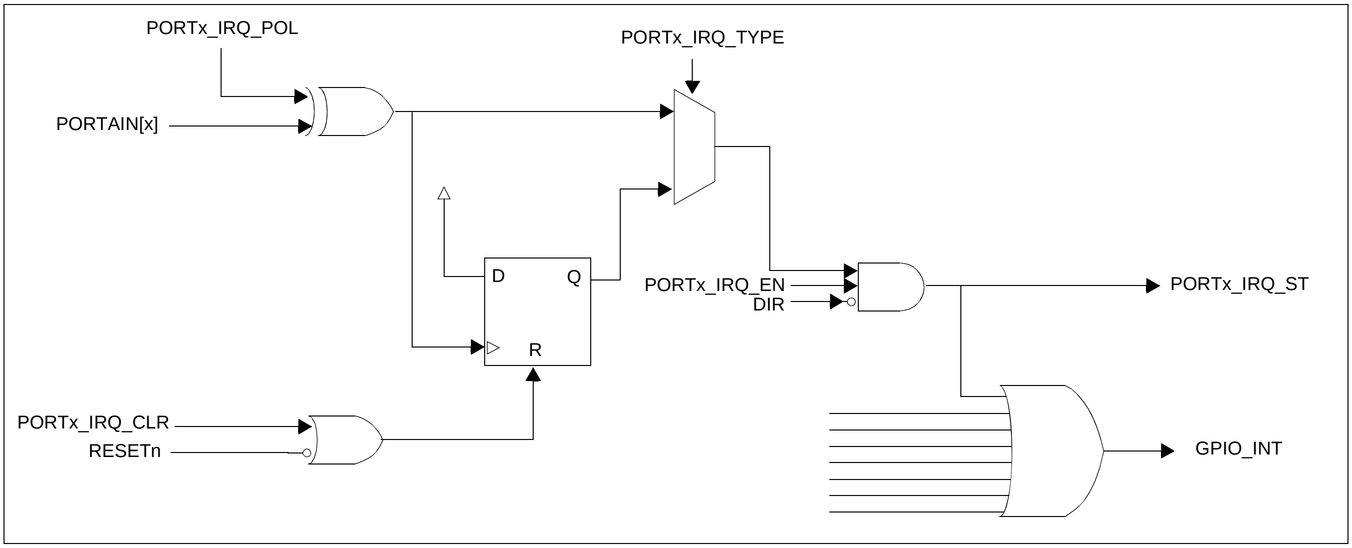


Figure 10-1: GPIOA and GPIOB Interrupt Logic

bits 15-8
GPIOB Interrupt Enable bits [7:0]
The GPIOB[7:0] pins can generate interrupts based on the setting of these bits. Each bit controls whether the corresponding GPIOB[7:0] pin has interrupts enabled. For example, setting bit 7 to 1b enables the interrupt for the GPIOB7 pin. The status of each interrupt is indicated by the bits in REG[0C2Ah]. Before enabling any interrupts, the Interrupt Type (REG[0C24h]) and Interrupt Polarity (REG[0C26h]) should be configured.
When this bit = 0b, the interrupt associated with GPIOBx is disabled.
When this bit = 1b, the interrupt associated with GPIOBx is enabled.

bits 7-0
GPIOA Interrupt Enable bits [7:0]
The GPIOA[7:0] pins can generate interrupts based on the setting of these bits. Each bit controls whether the corresponding GPIOA[7:0] pin has interrupts enabled. For example, setting bit 7 to 1b enables the interrupt for the GPIOA7 pin. The status of each interrupt is indicated by the bits in REG[0C2Ah]. Before enabling any interrupts, the Interrupt Type (REG[0C24h]) and Interrupt Polarity (REG[0C26h]) should be configured.
When this bit = 0b, the interrupt associated with GPIOAx is disabled.
When this bit = 1b, the interrupt associated with GPIOAx is enabled.

REG[0C2Ah] GPIOA&B IRQ Status and Clear Register Default = 0000h							Read/Write
GPIOB Interrupt Status bits 7-0							
15	14	13	12	11	10	9	8
GPIOA Interrupt Status bits 7-0							
7	6	5	4	3	2	1	0

bits 15-8 GPIOB Interrupt Status bits [7:0]
When an interrupt is enabled using REG[0C28h], these bits indicate the status of the corresponding GPIOB[7:0] interrupt. For example, reading bit 7 indicates the status of the GPIOB7 interrupt.
When this bit = 0b, an interrupt has not occurred.
When this bit = 1b, an interrupt has occurred.

To clear the interrupt status bit, write a 1b then a 0b to the corresponding bit.

bits 7-0 GPIOA Interrupt Status bits [7:0]
When an interrupt is enabled using REG[0C28h], these bits indicate the status of the corresponding GPIOA[7:0] interrupt. For example, reading bit 7 indicates the status of the GPIOA7 interrupt.
When this bit = 0b, an interrupt has not occurred.
When this bit = 1b, an interrupt has occurred.

To clear the interrupt status bit, write a 1b then a 0b to the corresponding bit.

10.4.7 Sprite Registers

The Sprite Registers are used in two different ways depending on the setting of the Image Format Converter Enable bit, REG[1700h] bit 8. The registers cannot be used for both functions at the same time.

When the Sprite Module is configured to Sprite Engine Mode (REG[1700h] bit 8 = 0b), REG[1000h] through REG[15FEh] are used by the Sprite Engine to define individual sprite characteristics for sprites #0 through #15. For this mode, refer to the register descriptions in Section , “Sprite Registers when Sprite Engine is Enabled” on page 215.

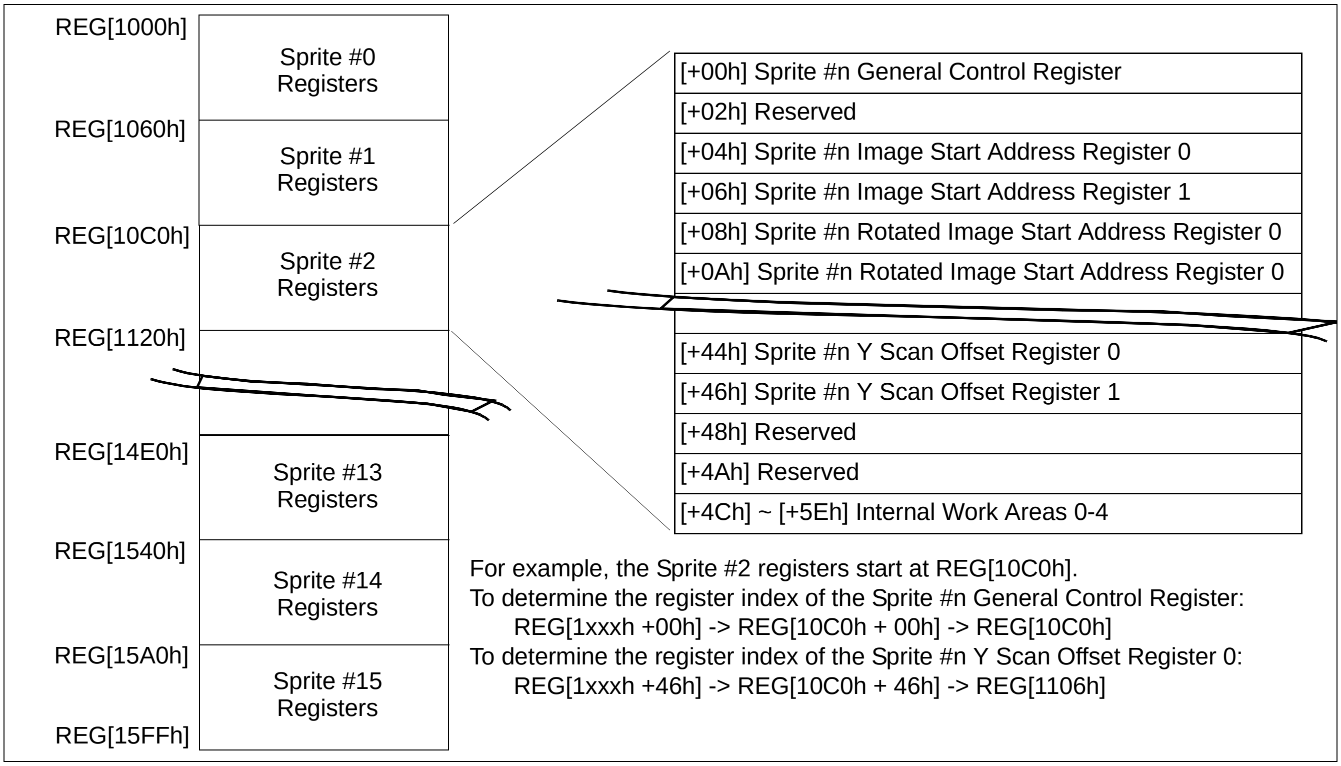
When the Sprite Module is configured to Image Format Converter Mode (REG[1700h] bit 8 = 1b), REG[1000h] through REG[100Ah] are used to specify the parameters required by the Image Format Converter. For this mode, refer to the register descriptions in Section , “Sprite Registers when Image Format Converter is Enabled” on page 231.

Sprite Registers when Sprite Engine is Enabled

When the Sprite Module is configured to Sprite Engine Mode (REG[1700h] bit 8 = 0b), these registers are used by the Sprite Engine to define individual sprite characteristics for sprites #0 through #15. These registers should be modified only when the Sprite Engine is idle, REG[1702h] bit 15 = 0b. For this mode, the same set of registers are repeated for each sprite and are mapped as follows. All Sprite #0~#15 registers begin with undefined values and must be programmed with the required values before enabling the sprite.

Note

REG[1000h]~REG[15FFh] consist of SRAM and should be initialized to 0000h before any sprite operations.



[+00h] Sprite #n General Control Register
[+02h] Reserved
[+04h] Sprite #n Image Start Address Register 0
[+06h] Sprite #n Image Start Address Register 1
[+08h] Sprite #n Rotated Image Start Address Register 0
[+0Ah] Sprite #n Rotated Image Start Address Register 0
[+44h] Sprite #n Y Scan Offset Register 0
[+46h] Sprite #n Y Scan Offset Register 1
[+48h] Reserved
[+4Ah] Reserved
[+4Ch] ~ [+5Eh] Internal Work Areas 0-4

For example, the Sprite #2 registers start at REG[10C0h].
To determine the register index of the Sprite #n General Control Register:
REG[1xxxh +00h] -> REG[10C0h + 00h] -> REG[10C0h]
To determine the register index of the Sprite #n Y Scan Offset Register 0:
REG[1xxxh +46h] -> REG[10C0h + 46h] -> REG[1106h]

Figure 10-2: Sprite #0 - #15 Register Mapping

REG[1xxxh +00h] Sprite #n General Control Register
Default = XXXXh Read/Write

Sprite #n Z-Order bits 3-0				Sprite #n Rotation bits 1-0		Sprite #n Mirror Enable	Sprite #n Transparency Enable
15	14	13	12	11	10	9	8
Sprite #n Arbitrary Rotation Enable	n/a					Sprite #n Image Format	Sprite #n Enable
7	6	5	4	3	2	1	0

bits 15-12
Sprite #n Z-Order bits [3:0]
These bits specify the Z-order associated with Sprite #n which determines the priority of the sprite for alpha blending and transparency functions. The Z-order value ranges from Fh which signifies the top (foreground) to 0h which signifies the bottom (background). If more than one sprite is assigned the same Z-order, the higher numbered sprite takes priority over the lower numbered sprite. Sprite #0 is used as the background and must be set to the lowest Z-order. Sprite #0 must have the same frame size as the display window (REG[1xxxh +10h] ~ REG[1xxxh +12h]).

bits 11-10

Sprite #n Rotation bits [1:0]
These bits specify the clockwise rotation applied to the Sprite #n image.

Table 10-45 : Sprite #n Rotation

REG[1xxxh +00h] bits 11-10	Sprite #n Rotation
00b	0° rotation
01b	90° rotation
10b	180° rotation
11b	270° rotation

Note

1. If arbitrary rotation is enabled (REG[1xxxh +00h] bit 7), these bits must be set to 00b (0° rotation).
2. If the frame size of the sprite is smaller than the virtual size, the top left portion of the image in memory is displayed. For 90° or 270°, the bottom left portion of the original non-rotated image is displayed since that part is the top left of the rotated image. For additional information, refer to
Figure 16-9: “Sprite Frame Size Less Than Sprite Virtual Size” on page 409.
3. If these bits are used for Sprite #0 (i.e. the background sprite is rotated), the rendering window of the sprite engine output will also be rotated.

bit 9

Sprite #n Mirror Enable
This bit controls the horizontal mirror function for Sprite #n.
When this bit = 0b, mirror is disabled.
When this bit = 1b, mirror is enabled.

bit 8

Sprite #n Transparency Enable
This bit controls the transparency function for Sprite #n. When a pixel is transparent, the next visible pixel below it, according to the Z-order, becomes visible.
When this bit = 0b, transparency is disabled.
When this bit = 1b, transparency is enabled.

Note

Sprite #0 (background) must not have any transparent pixels.

bit 7

Sprite #n Arbitrary Rotation Enable
This bit controls whether arbitrary rotation can be applied to Sprite #n. Arbitrary rotation should only be enabled when the Sprite #n Rotation bits (REG[1xxxh +00h] bits 11-10) are set to 00b (0° rotation).
When this bit = 0b, arbitrary rotation is disabled.
When this bit = 1b, arbitrary rotation is enabled.

bit 1 Sprite #n Image Data Format
This bit selects the data format of the Sprite #n image. Typically, stream data format is used when arbitrary rotation is not required. 8x8 tile data format should be selected when arbitrary rotation is enabled, REG[1xxxh +00h] bit 7 = 1b.
When this bit = 0b, stream data format is selected.
When this bit = 1b, 8x8 tile format is selected.

Note

It is recommended that the 8x8 tile format only be used for sprites with virtual image dimensions that are multiples of 8 and are to be used with arbitrary rotation.

bit 0 Sprite n# Enable
This bit controls the associated sprite (i.e. Sprite #n). A minimum of one sprite must be enabled before a sprite operation is triggered.
When this bit = 0b, the sprite is disabled.
When this bit = 1b, the sprite is enabled.

REG[1xxxh +02h] is Reserved

This register is Reserved and should not be written.

REG[1xxxh +04h] Sprite #n Image Start Address Register 0
Default = XXXXh Read/Write

Sprite #n Image Start Address bits 15-8							
15	14	13	12	11	10	9	8
Sprite #n Image Start Address bit 7	Reserved						
7	6	5	4	3	2	1	0

REG[1xxxh +06h] Sprite #n Image Start Address Register 1
Default = XXXXh Read/Write

Reserved					Sprite #n Image Start Address bits 26-24		
15	14	13	12	11	10	9	8
Sprite #n Image Start Address bits 23-16							
7	6	5	4	3	2	1	0

REG[1xxxh +06] bits 10-0
REG[1xxxh +04] bits 15-7
Sprite #n Image Start Address bits [26:7]
These bits specify bits 26-7 of the memory start address for the Sprite #n image when 0° or 180° rotation is selected. Each image start address must be aligned on a 256 byte boundary.

REG[1xxxh +04h] bits 6-0
Reserved
These bits must be set to 000_0000b.

REG[1xxxh +06h] bits 15-11
Reserved
These bits must be set to 0_0000b.

REG[1xxxh +08h] Sprite #n Rotated Image Start Address Register 0
Default = XXXXh Read/Write

Sprite #n Rotated Image Start Address bits 15-8							
15	14	13	12	11	10	9	8
Sprite #n Rotated Image Start Address bit 7	Reserved						
7	6	5	4	3	2	1	0

REG[1xxxh +0Ah] Sprite #n Rotated Image Start Address Register 1
Default = XXXXh Read/Write

Reserved					Sprite #n Rotated Image Start Address bits 26-24		
15	14	13	12	11	10	9	8
Sprite #n Rotated Image Start Address bits 23-16							
7	6	5	4	3	2	1	0

REG[1xxxh +0A] bits 10-0
REG[1xxxh +08] bits 15-7

Sprite #n Rotated Image Start Address bits [26:7]
These bits specify bits 26-7 of the memory start address for the Sprite #n rotated image when 90° or 270° rotation is selected. Each image start address must be aligned on a 256 byte boundary.

Note
If rotation is not used, a rotated image does not need to be stored.

REG[1xxxh +08h] bits 6-0

Reserved
These bits must be set to 000_0000b.

REG[1xxxh +0Ah] bits 15-11

Reserved
These bits must be set to 0_0000b.

REG[1xxxh +0Ch] Sprite #n X Position Register
Default = XXXXh Read/Write

Sprite #n X Position Sign bits 5-0					Sprite #n X Position bits 10-8		
15	14	13	12	11	10	9	8
Sprite #n X Position bits 7-0							
7	6	5	4	3	2	1	0

bits 15-11 Sprite #n X Position Sign bits [5:0]
These bits are the extended sign bits which determine if the X position is negative with respect to the top left corner. Only bit 15 is used.

bits 10-0 Sprite #n X Position bits [10:0]
These bits specify the X position of the sprite reference point for Sprite #n with respect to the top left corner of the display area. A negative position value allows the sprite to move off the display in any direction.

REG[1xxxh +0Eh] Sprite #n Y Position Register
Default = XXXXh Read/Write

Sprite #n Y Position Sign bits 5-0					Sprite #n Y Position bits 10-8		
15	14	13	12	11	10	9	8
Sprite #n Y Position bits 7-0							
7	6	5	4	3	2	1	0

bits 15-11 Sprite #n Y Position Sign bits [5:0]
These bits are the extended sign bits which determine if the Y position is negative with respect to the top left corner. Only bit 15 is used.

bits 10-0 Sprite #n Y Position bits [10:0]
These bits specify the Y position of the sprite reference point for Sprite #n with respect to the top left corner of the display area. A negative position value allows the sprite to move off the display in any direction.

REG[1xxxh +10h] Sprite #n Frame Width Register
Default = XXXXh Read/Write

n/a					Sprite #n Frame Width bits 10-8		
15	14	13	12	11	10	9	8
Sprite #n Frame Width bits 7-0							
7	6	5	4	3	2	1	0

bits 10-0 Sprite #n Frame Width bits [10:0]
These bits specify the width of the area in which the sprite is rendered, in pixels. These bits must be programmed such that the following formula is valid.
$1 \leq \text{Frame Width} \leq 1280$

Note

1. The frame width must be less than or equal to the virtual width unless arbitrary rotation is enabled, REG[1xxxh +00h] bit 7 = 1b.
2. If the frame size of the sprite is smaller than the virtual size, the top left portion of the image in memory is displayed. For 90° or 270°, the bottom left portion of the original non-rotated image is displayed since that part is the top left of the rotated image. For additional information, refer to
Figure 16-9: “Sprite Frame Size Less Than Sprite Virtual Size” on page 409.
3. The frame width of Sprite#0 must be the same size as the display area.
4. If REG[1xxx +00h] bits 11-10 are used to specify rotation for Sprite #0 (i.e. the background sprite is rotated), the rendering window of the sprite engine output will also be rotated.

REG[1xxxh +12h] Sprite #n Frame Height Register
Default = XXXXh Read/Write

n/a					Sprite #n Frame Height bits 10-8		
15	14	13	12	11	10	9	8
Sprite #n Frame Height bits 7-0							
7	6	5	4	3	2	1	0

bits 10-0 Sprite #n Frame Height bits [10:0]
These bits specify the height of the area in which the sprite is rendered, in pixels. These bits must be programmed such that the following formula is valid.

$1 \leq \text{Frame Height} \leq 1024$

Note

1. The frame height must be less than or equal to the virtual height unless arbitrary rotation is enabled, REG[1xxxh +00h] bit 7 = 1b.
2. If the frame size of the sprite is smaller than the virtual size, the top left portion of the image in memory is displayed. For 90° or 270°, the bottom left portion of the original non-rotated image is displayed since that part is the top left of the rotated image. For additional information, refer to
Figure 16-9: “Sprite Frame Size Less Than Sprite Virtual Size” on page 409.
3. The frame height of Sprite#0 must be the same size as the display area.
4. If REG[1xxx +00h] bits 11-10 are used to specify rotation for Sprite #0 (i.e. the background sprite is rotated), the rendering window of the sprite engine output will also be rotated.

REG[1xxxh +14h] Sprite #n Reference Point X Offset Register
Default = XXXXh Read/Write

Sprite #n Reference Point X Offset Sign bits 5-0					Sprite #n Reference Point X Offset bits 10-8		
15	14	13	12	11	10	9	8
Sprite #n Reference Point X Offset bits 7-0							
7	6	5	4	3	2	1	0

bits 15-11 Sprite #n Reference Point X Offset Sign bits [5:0]
These bits are the extended sign bits which determine if the Reference Point X Offset is negative with respect to the top left corner of the sprite. Only bit 15 is used.

bits 10-0 Sprite #n Reference Point X Offset bits [10:0]
These bits specify the Reference Point X Offset with respect to the top left corner of the sprite.

Note

Once set, the reference point can be used to set and query the location of the sprite. The reference point also serves as the “center” for all transforms (rotation and mirror). If desired, the reference point may be defined outside of the sprite’s bounds.

REG[1xxxh +16h] Sprite #n Reference Point Y Offset Register
Default = XXXXh Read/Write

Sprite #n Reference Point Y Offset Sign bits 5-0					Sprite #n Reference Point Y Offset bits 10-8		
15	14	13	12	11	10	9	8
Sprite #n Reference Point X Offset bits 7-0							
7	6	5	4	3	2	1	0

bits 15-11 Sprite #n Reference Point Y Offset Sign bits [5:0]
These bits are the extended sign bits which determine if the Reference Point Y Offset is negative with respect to the top left corner of the sprite. Only bit 15 is used.

bits 10-0 Sprite #n Reference Point Y Offset bits [10:0]
These bits specify the Reference Point Y Offset with respect to the top left corner of the sprite.

Note
Once set, the reference point can be used to set and query the location of the sprite. The reference point also serves as the "center" for all transforms (rotation and mirror). If desired, the reference point may be defined outside of the sprite's bounds.

REG[1xxxh +18h] Sprite #n Transparency Color/Texture Alpha Register
Default = XXXXh Read/Write

Sprite #n Transparency Color / Texture Alpha bits 15-8							
15	14	13	12	11	10	9	8
Sprite #n Transparency Color / Texture Alpha bits 7-0							
7	6	5	4	3	2	1	0

bits 15-0 Sprite #n Transparency Color / Texture Alpha bits [15:0]
These bits specify the transparency color / texture alpha for the sprite. For RGB 5:6:5, the transparency color is defined using bits 15-0. If the sprite data is ARGB 1:5:5:5, these bits define the alpha value based on the alpha index value as shown in the following table

Table 10-46 : Transparency Color/Texture Alpha

REG[1xxxh +18h]	Image		
	RGB 5:6:5	ARGB 1:5:5:5	ARGB 4:4:4:4 / ARGB 8:8:8:8
bits 15-8	Transparent Pixel Code (16-bit)	Alpha Value when A=1 (8-bit)	Invalid
bits 7-0		Alpha Value when A=0 (8-bit)	

<table>
<tr><td colspan="8"><b>REG[1xxxh +1Ah] Sprite #n Color Format Register</b>
Default = XXXXh Read/Write</td></tr>
<tr><td colspan="8">n/a</td></tr>
<tr><td>15</td><td>14</td><td>13</td><td>12</td><td>11</td><td>10</td><td>9</td><td>8</td></tr>
<tr><td colspan="6">n/a</td><td colspan="2">Sprite #n Color Format bits 1-0</td></tr>
<tr><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
</table>

bits 1-0 Sprite #n Color Format bits [1:0]
These bits determine the color format for the sprite.

Table 10-47 : Sprite #n Color Format Selection

REG[1xxxh +1Ah]	Color Format	Sprite Transparency Color
00b	RGB 5:6:5 (16 bpp)	Transparency Color (see REG[1xxxh +18h])
01b	ARGB 1:5:5:5 (16 bpp)	Texture Alpha (see REG[1xxxh +18h])
10b	ARGB 4:4:4:4 (16 bpp)	Unused
11b	ARGB 8:8:8:8 (32 bpp)	Unused

REG[1xxxh +1Ch] Sprite #n Frame Sequence Register 0
Default = XXXXh Read/Write

Sprite #n Sequence 1 Frame Number bits 7-0							
15	14	13	12	11	10	9	8
Sprite #n Sequence 0 Frame Number bits 7-0							
7	6	5	4	3	2	1	0

REG[1xxxh +1Eh] Sprite #n Frame Sequence Register 1
Default = XXXXh Read/Write

Sprite #n Sequence 3 Frame Number bits 7-0							
15	14	13	12	11	10	9	8
Sprite #n Sequence 2 Frame Number bits 7-0							
7	6	5	4	3	2	1	0

REG[1xxxh +20h] Sprite #n Frame Sequence Register 2
Default = XXXXh Read/Write

Sprite #n Sequence 5 Frame Number bits 7-0							
15	14	13	12	11	10	9	8
Sprite #n Sequence 4 Frame Number bits 7-0							
7	6	5	4	3	2	1	0

REG[1xxxh +22h] Sprite #n Frame Sequence Register 3
Default = XXXXh Read/Write

Sprite #n Sequence 7 Frame Number bits 7-0							
15	14	13	12	11	10	9	8
Sprite #n Sequence 6 Frame Number bits 7-0							
7	6	5	4	3	2	1	0

REG[1xxxh +24h] Sprite #n Frame Sequence Register 4
Default = XXXXh Read/Write

Sprite #n Sequence 9 Frame Number bits 7-0							
15	14	13	12	11	10	9	8
Sprite #n Sequence 8 Frame Number bits 7-0							
7	6	5	4	3	2	1	0

REG[1xxxh +26h] Sprite #n Frame Sequence Register 5
Default = XXXXh Read/Write

Sprite #n Sequence 11 Frame Number bits 7-0							
15	14	13	12	11	10	9	8
Sprite #n Sequence 10 Frame Number bits 7-0							
7	6	5	4	3	2	1	0

REG[1xxxh +28h] Sprite #n Frame Sequence Register 6
Default = XXXXh Read/Write

Sprite #n Sequence 13 Frame Number bits 7-0							
15	14	13	12	11	10	9	8
Sprite #n Sequence 12 Frame Number bits 7-0							
7	6	5	4	3	2	1	0

REG[1xxxh +2Ah] Sprite #n Frame Sequence Register 7
Default = XXXXh Read/Write

Sprite #n Sequence 15 Frame Number bits 7-0							
15	14	13	12	11	10	9	8
Sprite #n Sequence 14 Frame Number bits 7-0							
7	6	5	4	3	2	1	0

REG[1xxxh +2A] bits 15-8
REG[1xxxh +2A] bits 7-0
REG[1xxxh +28] bits 15-8
REG[1xxxh +28] bits 7-0
REG[1xxxh +26] bits 15-8
REG[1xxxh +26] bits 7-0
REG[1xxxh +24] bits 15-8
REG[1xxxh +24] bits 7-0
REG[1xxxh +22] bits 15-8
REG[1xxxh +22] bits 7-0
REG[1xxxh +20] bits 15-8
REG[1xxxh +20] bits 7-0
REG[1xxxh +1E] bits 15-8
REG[1xxxh +1E] bits 7-0
REG[1xxxh +1C] bits 15-8
REG[1xxxh +1C] bits 7-0

Sprite #n Sequence X Frame Number bits [7:0]
When the Sprite #n sequence is disabled (see REG[1780h] ~ REG[179Eh]), these bits are ignored.
When the Sprite #n sequence is enabled (see REG[1780h] ~ REG[179Eh]), these bits determine the index into memory of the frame used for sequence X. For example if the Sprite #n Sequence 0 Frame Number bits = 0000_0111b, frame 7 is used for sequence 0.

Note

For REG[1782h] ~ REG[179Eh] bits 1-0 = 10b, REG[1xxxh +1Ch] bits 7-0 indicate the frame number which will be used for the next sprite operation.

REG[1xxxh +2Ch] Sprite #n Virtual Image Width Register							
Default = XXXXh							Read/Write
n/a		Sprite #n Virtual Image Width bits 13-8					
15	14	13	12	11	10	9	8
Sprite #n Virtual Image Width bits 7-0							
7	6	5	4	3	2	1	0

bits 13-0 Sprite #n Virtual Image Width bits [13:0]
These bits specify the horizontal size (or width) of the sprite virtual image, in pixels. The virtual image can be larger than the actual display area.

Note

These bits must be programmed such that the following formula is valid.

$1 \leq$ Virtual Image Width ≤ 8192

REG[1xxxh +2Eh] Sprite #n Virtual Image Height Register							
Default = XXXXh							Read/Write
n/a		Sprite #n Virtual Image Height bits 13-8					
15	14	13	12	11	10	9	8
Sprite #n Virtual Image Height bits 7-0							
7	6	5	4	3	2	1	0

bits 13-0 Sprite #n Virtual Image Height bits [13:0]
These bits specify the vertical size (or height) of the sprite virtual image, in pixels. The virtual image can be larger than the actual display area.

Note

These bits must be programmed such that the following formula is valid.

$1 \leq$ Virtual Image Width ≤ 8192

REG[1xxxh +30h] Sprite #n X Scan Vector H Register 0
Default = XXXXh Read/Write

Sprite #n X Scan Vector H bits 15-8							
15	14	13	12	11	10	9	8
Sprite #n X Scan Vector H bits 7-0							
7	6	5	4	3	2	1	0

REG[1xxxh +32h] Sprite #n X Scan Vector H Register 1
Default = XXXXh Read/Write

n/a							
15	14	13	12	11	10	9	8
n/a	Sprite #n X Scan Vector H bits 22-16						
7	6	5	4	3	2	1	0

REG[1xxxh +32h] bits 6-0
REG[1xxxh +30h] bits 15-0

Sprite #n X Scan Vector H bits [22:0]
These bits specify the X Scan Vector H bits for the Sprite, in [1. 13. 9] S.I.F format. These bits are used for arbitrary angle rotation of the sprite, see Section 16.7, “Arbitrary Angle Rotation” on page 414.

Note
[1,13,9] S.I.F. includes a 1-bit sign, 13-bit integer value, and 9-bit fractional value.

REG[1xxxh +34h] Sprite #n Y Scan Vector H Register 0
Default = XXXXh Read/Write

Sprite #n Y Scan Vector H bits 15-8							
15	14	13	12	11	10	9	8
Sprite #n Y Scan Vector H bits 7-0							
7	6	5	4	3	2	1	0

REG[1xxxh +36h] Sprite #n Y Scan Vector H Register 1
Default = XXXXh Read/Write

n/a							
15	14	13	12	11	10	9	8
n/a	Sprite #n Y Scan Vector H bits 22-16						
7	6	5	4	3	2	1	0

REG[1xxxh +36h] bits 6-0
REG[1xxxh +34h] bits 15-0

Sprite #n Y Scan Vector H bits [22:0]
These bits specify the Y Scan Vector H bits for the Sprite, in [1. 13. 9] S.I.F format. These bits are used for arbitrary angle rotation of the sprite, see Section 16.7, “Arbitrary Angle Rotation” on page 414.

Note
[1,13,9] S.I.F. includes a 1-bit sign, 13-bit integer value, and 9-bit fractional value.

REG[1xxxh +38h] Sprite #n X Scan Vector V Register 0							
Default = XXXXh							Read/Write
Sprite #n X Scan Vector V bits 15-8							
15	14	13	12	11	10	9	8
Sprite #n X Scan Vector V bits 7-0							
7	6	5	4	3	2	1	0

REG[1xxxh +3Ah] Sprite #n X Scan Vector V Register 1							
Default = XXXXh							Read/Write
n/a							
15	14	13	12	11	10	9	8
n/a	Sprite #n X Scan Vector V bits 22-16						
7	6	5	4	3	2	1	0

REG[1xxxh +3Ah] bits 6-0
REG[1xxxh +38h] bits 15-0

Sprite #n X Scan Vector V bits [22:0]
These bits specify the X Scan Vector V bits for the Sprite, in [1. 13. 9] S.I.F format. These bits are used for arbitrary angle rotation of the sprite, see Section 16.7, “Arbitrary Angle Rotation” on page 414.

Note

[1,13,9] S.I.F. includes a 1-bit sign, 13-bit integer value, and 9-bit fractional value.

REG[1xxxh +3Ch] Sprite #n Y Scan Vector V Register 0							
Default = XXXXh							Read/Write
Sprite #n Y Scan Vector V bits 15-8							
15	14	13	12	11	10	9	8
Sprite #n Y Scan Vector V bits 7-0							
7	6	5	4	3	2	1	0

REG[1xxxh +3Eh] Sprite #n Y Scan Vector V Register 1							
Default = XXXXh							Read/Write
n/a							
15	14	13	12	11	10	9	8
n/a	Sprite #n Y Scan Vector V bits 22-16						
7	6	5	4	3	2	1	0

REG[1xxxh +3Eh] bits 6-0
REG[1xxxh +3Ch] bits 15-0

Sprite #n Y Scan Vector V bits [22:0]
These bits specify the Y Scan Vector V bits for the Sprite, in [1. 13. 9] S.I.F format. These bits are used for arbitrary angle rotation of the sprite, see Section 16.7, “Arbitrary Angle Rotation” on page 414.

Note

[1,13,9] S.I.F. includes a 1-bit sign, 13-bit integer value, and 9-bit fractional value.

REG[1xxxh +40h] Sprite #n X Scan Offset Register 0
Default = XXXXh
							Read/Write

Sprite #n X Scan Offset bits 15-8							
15	14	13	12	11	10	9	8
Sprite #n X Scan Offset bits 7-0							
7	6	5	4	3	2	1	0

REG[1xxxh +42h] Sprite #n X Scan Offset Register 1
Default = XXXXh
							Read/Write

Sprite #n X Scan Offset bits 31-24							
15	14	13	12	11	10	9	8
Sprite #n X Scan Offset bits 23-16							
7	6	5	4	3	2	1	0

REG[1xxxh +42h] bits 15-0
REG[1xxxh +40h] bits 15-0

Sprite #n X Scan Offset bits [31:0]
These bits specify the X Scan Offset bits for the Sprite, in [1. 13. 9] S.I.F format. These bits are used for arbitrary angle rotation of the sprite, see Section 16.7, “Arbitrary Angle Rotation” on page 414.

Note
[1,13,9] S.I.F. includes a 1-bit sign, 13-bit integer value, and 9-bit fractional value.

REG[1xxxh +44h] Sprite #n Y Scan Offset Register 0
Default = XXXXh
							Read/Write

Sprite #n Y Scan Offset bits 15-8							
15	14	13	12	11	10	9	8
Sprite #n Y Scan Offset bits 7-0							
7	6	5	4	3	2	1	0

REG[1xxxh +46h] Sprite #n Y Scan Offset Register 1
Default = XXXXh
							Read/Write

Sprite #n Y Scan Offset bits 31-24							
15	14	13	12	11	10	9	8
Sprite #n Y Scan Offset bits 23-16							
7	6	5	4	3	2	1	0

REG[1xxxh +46h] bits 15-0
REG[1xxxh +44h] bits 15-0

Sprite #n Y Scan Offset bits [31:0]
These bits specify the Y Scan Offset bits for the Sprite, in [1. 13. 9] S.I.F format. These bits are used for arbitrary angle rotation of the sprite, see Section 16.7, “Arbitrary Angle Rotation” on page 414.

Note
[1,13,9] S.I.F. includes a 1-bit sign, 13-bit integer value, and 9-bit fractional value.

REG[1xxxh +48h] through REG[1xxxh +5Eh] are Reserved

These registers are Reserved and should not be written.

Sprite Registers when Image Format Converter is Enabled

When the Image Format Converter (IFC) is enabled (REG[1700h] bit 8 = 1b), these registers are used to specify some of the parameters required by the Image Format Converter.

REG[1004h] IFC Source Image Address Register 0 Default = 0000h							Read/Write
IFC Source Image Address bits 15-8							
15	14	13	12	11	10	9	8
IFC Source Image Address bits 7-1							n/a
7	6	5	4	3	2	1	0

REG[1006h] IFC Source Image Address Register 1 Default = 0000h							Read/Write
n/a					IFC Source Image Address bits 26-24		
15	14	13	12	11	10	9	8
IFC Source Image Address bits 23-16							
7	6	5	4	3	2	1	0

REG[1006h] bits 10-0
REG[1004h] bits 15-1 IFC Source Image Address bits [26:1]

When the IFC (Image Format Converter) is enabled (REG[1700h] bit 8 = 1b), these bits specify the source image address for the image to be converted. The IFC Source Image Address must be dword aligned (bits 1-0 must be set to 00b).

IFC Destination Image Address (see REG[1710h] ~ REG[1712h])

The registers used to control the Image Format Converter (IFC) destination address are located in REG[1710h] ~ REG[1712h]. For a detailed bit description, refer to the Sprite Frame Buffer 0 Start Address registers in Section 10.4.8, “Sprite Engine Registers” on page 233.

REG[102Ch] IFC Image Width Register
Default = 0000h Read/Write

n/a		IFC Image Width bits 13-8					
15	14	13	12	11	10	9	8
IFC Image Width bits 7-0							
7	6	5	4	3	2	1	0

bits 13-0 IFC Image Width bits [13:0]
When the IFC (Image Format Converter) is enabled (REG[1700h] bit 8 = 1b), these bits specify the width of the image to be converted. This value must be programmed such that the following formula is valid.

$1 \leq \text{image width} \leq 8192$

Note
The maximum supported horizontal size is 8192 pixels for 16 bpp and 2048 pixels for 32 bpp.

REG[102Eh] IFC Image Height Register
Default = 0000h Read/Write

n/a		IFC Image Height bits 13-8					
15	14	13	12	11	10	9	8
IFC Image Height bits 7-0							
7	6	5	4	3	2	1	0

bits 13-0 IFC Image Height bits [13:0]
When the IFC (Image Format Converter) is enabled (REG[1700h] bit 8 = 1b), these bits specify the height of the image to be converted. This value must be programmed such that the following formula is valid.

$1 \leq \text{image height} \leq 8192$

10.4.8 Sprite Engine Registers

REG[1700h] Sprite Control Register
Default = 0000h Read/Write

Sprite Software Reset	Reserved	n/a					Image Format Converter Enable
15	14	13	12	11	10	9	8
Sprite Individual Color Format Enable	n/a	Sprite Common Color Format bits 1-0		Frame Buffer Color Format bits 1-0		Double Frame Buffer Enable	Sprite Module Enable
7	6	5	4	3	2	1	0

bit 15

Sprite Software Reset
This bit performs a software reset of the Sprite module and resets the following registers to their default values: REG[1700h] ~ REG[1716h], and REG[1780h] ~ REG[179Eh].
This bit does not clear the Sprite registers (REG[1xxxh +00h] ~ REG[1xxxh +5Eh]).
When this bit = 0b, there is no hardware effect.
When this bit = 1b, the sprite module is reset.

bit 14

Reserved
The default value for this bit is 0b.

bit 8

Image Format Converter Enable
This bit selects which operational mode the Sprite module is configured for: Sprite Engine Mode or Image Format Converter (IFC) Mode. This bit does not activate the selected mode.
When this bit = 0b, Sprite Engine Mode is selected. To activate the Sprite Engine, write a 1b to the Sprite Module Enable bit, REG[1700h] bit 0 = 1b.
When this bit = 1b, Image Format Converter Mode is selected. To activate the Image Format Converter (IFC), write a 1b to the Manual Trigger bit, REG[1704h] bit 0 = 1b.

Note

Both REG[1700h] bit 8 and REG[1700h] bit 0 must be set to 1b in order to enable the IFC.

bit 7

Sprite Individual Color Format Enable
This bit determines whether all sprites share the same color format as specified by the Sprite Common Color Format bits (REG[1700h] bits 5-4), or each sprite has a specific color format associated with it.
When this bit = 0b, all sprites share the same color format as specified by the Sprite Common Color Format bits (REG[1700h] bits 5-4).
When this bit = 1b, each sprite (maximum of 16) has an individual color format specified by the Sprite #n Color Format register (see REG[1xxxh +1Ah]).

bits 5-4 Sprite Common Color Format bits [1:0]
These bits specify the color format associated with all sprites when the Sprite Individual Color Format bit is set to 0b. These bits are not used when individual color formats are selected (REG[1700h] bit 7 = 1b).

Table 10-48 : Sprite Common Color Format Selection

REG[1700h] bits 5-4	Color Format
00b (default)	RGB 5:6:5 (16 bpp)
01b	ARGB 1:5:5:5 (16 bpp)
10b	ARGB 4:4:4:4 (16 bpp)
11b	ARGB 8:8:8:8 (32 bpp)

Note

When Image Format Converter mode is selected (REG[1700h] bit 8 = 1b), these bits are used to distinguish between 16 bpp and 32 bpp.

bits 3-2 Frame Buffer Color Format bits [1:0]
These bits determine the color format of the “synthesized” pixel when it is written into the display buffer.

Table 10-49 : Frame Buffer Color Format Selection

REG[1700h] bits 3-2	Color Format
00b (default)	RGB 5:6:5 (16 bpp)
01b	RGB 3:3:2 (8 bpp)
10b	RGB 8:8:8 (24 bpp)
11b	Reserved

bit 1 Double Frame Buffer Enable
This bit controls the number of frame buffers used by the Sprite engine for rendering sprite frames. When double frame buffer mode is selected, two frame buffers are used as specified by the Sprite Frame Buffer 0 Start Address (REG[1710h] ~ REG[1712h]) and Sprite Frame Buffer 1 Start Address (REG[1714h] ~ REG[1716h]). When single frame buffer mode is selected, only one frame buffer is used as specified by the Sprite Frame Buffer 0 Start Address (REG[1710h] ~ REG[1712h]).
When this bit = 0b, single frame buffer mode is selected. (default)
When this bit = 1b, double frame buffer mode is selected.

bit 0 Sprite Module Enable
This bit controls the Sprite module.
When this bit = 0b, the sprite module is disabled and the Sprite/Sprite Engine/IFC registers must not be written to. (default)
When this bit = 1b, the sprite module is enabled and the sprite engine is activated when the Image Format Converter Enable bit is set to 0b (REG[1700h] bit 8 = 0b).

Note

Both REG[1700h] bit 8 and REG[1700h] bit 0 must be set to 1b in order to enable the IFC.

REG[1702h] Sprite Status Register Default = 8000h							Read Only
Sprite Status	n/a						
15	14	13	12	11	10	9	8
n/a						Reserved	n/a
7	6	5	4	3	2	1	0

bit 15 Sprite Status (Read Only)
This bit indicates the status of the sprite module. The sprite module becomes idle when the current sprite operation is complete. It becomes busy again when a trigger for a new operation is received. The idle time varies based on how many sprites are enabled, sprite sizes, trigger mode, refresh rate, etc.
When this bit = 0b, the sprite module is idle.
When this bit = 1b, a sprite operation is being performed.

Note

The Sprite registers (REG[1xxxh +00h] ~ REG[1xxxh +46h]) should be written to only when the sprite engine is idle, REG[1702h] bit 15 = 0b.

bit 1 Reserved
The default value for this bit is 0b.

REG[1704h] Sprite Frame Sequence Trigger Control Register							
Default = 0000h							Read/Write
n/a					Sprite Trigger Mode bits 2-0		
15	14	13	12	11	10	9	8
Sprite Number bits 3-0				n/a	Previous Frame	Next Frame	Manual Trigger
7	6	5	4	3	2	1	0

bits 10-8 Sprite Trigger Mode bits [2:0]

These bits control how sprite operations are triggered.

Table 10-50 : Sprite Trigger Mode Selection

REG[1704h] bits 10-8	Sprite Trigger Mode
000b	Sprite operations are triggered manually using the Manual Trigger bit, REG[1704h] bit 0.
001b	Every 1 VSYNC can trigger a new sprite operation.
010b	Every 2 VSYNCs can trigger a new sprite operation.
011b	Every 3 VSYNCs can trigger a new sprite operation.
100b	Every 4 VSYNCs can trigger a new sprite operation.
101b	Every 5 VSYNCs can trigger a new sprite operation.
110b	Reserved
111b	Reserved

Note

1. If sprite operations are set to trigger every 1 to 5 VSYNCs, the Sprite Interrupt Raw Status must be cleared before the next sprite operation takes place (REG[1708h] bit 1 = 1b).
2. If sprite operations are set to trigger every 1 to 5 VSYNCs, the actual sprite operation may take longer than the configured interval, especially for high LCDDCLK configurations.

bits 7-4 Sprite Number bits [3:0]

When manual sprite triggering is selected (REG[1704h] bits 10-8 = 000b), these bits specify which sprite (out of 16 possible sprites) is controlled manually. Once the sprite number is selected, the frame sequence can be controlled using the Previous Frame bit (REG[1704h] bit 2), Next Frame bit (REG[1704h] bit 1), and the Manual Trigger bit (REG[1704h] bit 0.

bit 2 Previous Frame
When manual sprite triggering is selected (REG[1704h] bits 10-8 = 000b), this bit decrements to the previous frame of the sprite specified by the Sprite Number bits, REG[1704h] bits 7-4.
When this bit = 0b, the frame is not changed.
When this bit = 1b, the frame sequence number for the specified sprite (see REG[1704h] bits 7-4) is decremented (-1).

Note

The Previous Frame and Manual Trigger bits must not be set at the same time. For example, setting REG[1704h] = 00x5h is not valid.

bit 1 Next Frame
When manual sprite triggering is selected (REG[1704h] bits 10-8 = 000b), this bit increments to the next frame of the sprite specified by the Sprite Number bits, REG[1704h] bits 7-4.
When this bit = 0b, the frame is not changed.
When this bit = 1b, the frame sequence number for the specified sprite (see REG[1704h] bits 7-4) is incremented (+1).

Note

The Next Frame and Manual Trigger bits must not be set at the same time. For example, setting REG[1704h] = 00x3h is not valid.

bit 0 Manual Trigger
When manual sprite triggering is selected (REG[1704h] bits 10-8 = 000b), this bit is used to manually trigger a new sprite operation.
When this bit = 0b, a sprite operation is not triggered.
When this bit = 1b, a new sprite operation is triggered.

Note

1. When Image Format Converter (IFC) Mode is selected (REG[1700h] bit 8 = 1b), the image converter function starts when this bit is set to 1b.
2. When an image format conversion is performed, confirm that the conversion is complete (see REG[1704h] bit 1) before starting a new image format conversion.
3. The Next Frame/Previous Frame bits must not be set at the same time as the Manual Trigger bit. For example, setting REG[1704h] = 00x3h or 00x5h is not valid.

REG[1706h] Sprite Interrupt Control Register
Default = 0000h Read/Write

n/a							
15	14	13	12	11	10	9	8
n/a						Sprite Interrupt Enable	n/a
7	6	5	4	3	2	1	0

bit 1 Sprite Interrupt Enable
This bit controls the Sprite Interrupt which occurs when the Sprite operation or Image Format Conversion has completed.
When this bit = 0b, the sprite interrupt is disabled.
When this bit = 1b, the sprite interrupt is enabled.

REG[1708h] Sprite Interrupt Status Register
Default = 0000h Read/Write

n/a							
15	14	13	12	11	10	9	8
n/a						Sprite Interrupt Raw Status	n/a
7	6	5	4	3	2	1	0

bit 1 Sprite Interrupt Raw Status
This bit indicates the raw status of the Sprite Interrupt which occurs when the Sprite operation or Image Format Conversion has completed.
When this bit = 0b, a sprite interrupt has not occurred.
When this bit = 1b, a sprite interrupt has occurred.

To clear this status bit, write a 1b to this bit.

Note

If sprite operations are set to trigger every 1 to 5 VSYNCs (REG[1704h] bits 10-8 = 001b-101b), the Sprite Interrupt Raw Status must be cleared before the next sprite operation takes place.

REG[1710h] Sprite Frame Buffer 0 Start Address Register 0 Default = 0000h							Read/Write
Sprite Frame Buffer 0 Start Address bits 15-8							
15	14	13	12	11	10	9	8
Sprite Frame Buffer 0 Start Address bits 7-3					n/a		
7	6	5	4	3	2	1	0

REG[1712h] Sprite Frame Buffer 0 Start Address Register 1 Default = 0000h							Read/Write
n/a					Sprite Frame Buffer 0 Start Address bits 26-24		
15	14	13	12	11	10	9	8
Sprite Frame Buffer 0 Start Address bits 23-16							
7	6	5	4	3	2	1	0

This register is used in different ways depending on whether the Sprite Engine or Image Format Converter (IFC) is enabled (see REG[1700h] bit 8).

For Sprite Engine Enabled (REG[1700h] bit 8 = 0b)

REG[1712h] bits 10-0
REG[1710h] bits 15-3 Sprite Frame Buffer 0 Start Address bits [26:3]

These bits specify bits 26-3 of the memory start address for Sprite Frame Buffer 0 which is used for Sprite operations. Sprite Frame Buffer 0 is used for both double frame buffer mode and single frame buffer mode (see REG[1700h] bit 1).

For Image Format Converter (IFC) Enabled (REG[1700h] bit 8 = 1b)

REG[1712h] bits 10-0
REG[1710h] bits 15-7 IFC Destination Image Address bits [26:7]

These bits specify the destination image address for the resulting image that is to be converted. For 16 bpp mode, bits 26-7 are used (bits 6-0 must be set to 000_0000b). For 32 bpp mode, only bits 26-8 are used (bits 7-0 must be set to 0000_0000b).

After the image format conversion is complete, these bits are automatically updated. The address where it meets the requirement such as just behind the Image data that Convert is done the value after it updates it, and the above-mentioned invalid bit field =0 is indicated. It only has to set this register in front of Convert first time at doing Convert it only once. And, Read only has to do this register immediately after completion of each Convert if the first address of each individual Image that Convert is done is necessary

REG[1714h] Sprite Frame Buffer 1 Start Address Register 0

Default = 0000h Read/Write

Sprite Frame Buffer 1 Start Address bits 15-8							
15	14	13	12	11	10	9	8
Sprite Frame Buffer 1 Start Address bits 7-3					n/a		
7	6	5	4	3	2	1	0

REG[1716h] Sprite Frame Buffer 1 Start Address Register 1

Default = 0000h Read/Write

n/a					Sprite Frame Buffer 1 Start Address bits 26-24		
15	14	13	12	11	10	9	8
Sprite Frame Buffer 1 Start Address bits 23-16							
7	6	5	4	3	2	1	0

REG[1716h] bits 10-0
REG[1714h] bits 15-3 Sprite Frame Buffer 1 Start Address bits [26:3]

These bits specify bits 26-3 of the memory start address for Sprite Frame Buffer 1 which is used for Sprite operations. Sprite Frame Buffer 1 is only used when double frame buffer mode is selected, REG[1700h] bit 1 = 1b.

REG[1718h] is Reserved

This register is Reserved and should not be written.

REG[1750h] through REG[1774h] are Reserved

These registers are Reserved and should not be written.

REG[1780h] ~ REG[179Eh] Sprite #0-15 Frame Sequence Control Registers Default = 0000h							Read/Write
Sprite #0-15 Frame Sequence Length bits 3-0				Sprite #0-15 Frame Index bits 3-0			
15	14	13	12	11	10	9	8
n/a			Sprite #0-15 Operation VSYNC Trigger Count bits 2-0			Sprite #0-15 Frame Sequence Mode bits 1-0	
7	6	5	4	3	2	1	0

The following registers are used for frame sequence control for the corresponding Sprite #. The bit descriptions below refer to the as Sprite #n where n is 0-15 according to the Sprite # that needs to be controlled.

Table 10-51 : Sprite #0-15 Frame Sequence Control Register Summary

Register	Corresponding Sprite	Register	Corresponding Sprite
REG[1780h]	Sprite #0	REG[1790h]	Sprite #8
REG[1782h]	Sprite #1	REG[1792h]	Sprite #9
REG[1784h]	Sprite #2	REG[1794h]	Sprite #10
REG[1786h]	Sprite #3	REG[1796h]	Sprite #11
REG[1788h]	Sprite #4	REG[1798h]	Sprite #12
REG[178Ah]	Sprite #5	REG[179Ah]	Sprite #13
REG[178Ch]	Sprite #6	REG[179Ch]	Sprite #14
REG[178Eh]	Sprite #7	REG[179Eh]	Sprite #15

bits 15-12 Sprite #n Frame Sequence Length bits [3:0]
When the Sprite #n Frame Sequence Mode bits equal 01b (bits 1-0 = 01b), these bits specify the length of the frame sequence for Sprite #n.

REG[17xxh] bits 15-12 = total number of frame in the sequence - 1

bits 11-8 Sprite #n Frame Index bits [3:0]
These bits indicate the frame index that will be rendered for Sprite #n at the next sprite operation trigger. Reading from this register will also indicate the frame index for the current frame if VSync triggered sprite operations are not being used, and the frame index has not been changed for this sprite by the Previous Frame/Next Frame bits in REG[1704h] bits 2-1.

Note
When REG[1780h] ~ REG[179Eh] bits 1-0 = 10b, this field is treated as invalid.

bits 4-2 Sprite #n Operation VSYNC Trigger Count bits [2:0]
These bits determine how many VSYNC sprite operation triggers are required before the next frame is drawn. When these bits are written, the next sprite operation trigger paints the current frame index again.

REG[17xxh] bits 4-2 = total number of VSYNC sprite operation triggers - 1

bits 1-0 Sprite #n Frame Sequence Mode bits [1:0]
These bits select the frame sequence mode for Sprite #n.

Table 10-52 : Sprite #n Frame Sequence Mode Selection

REG[17xxh] bits 1-0	Frame Sequence Mode Selection
00b	Frame Sequence number is invalid. Always display the pattern specified by the Sprite #n Sequence 0 Frame Number bits (REG[1xxxh +1Ch] bits 7-0).
01b	Frame Sequence is valid. Update the Sprite #n Current Frame Index bits (REG[17xxh] bits 11-8) at every frame sequence update timing.
10b	Frame increment is valid. At every frame sequence update timing, increment the Sprite #n Sequence 0 Frame Number bits in REG[1xxxh +1Ch] bits 7-0 using the Sprite #n Sequence 2 Frame Number in REG[1xxxh +1Eh] bits 7-0 as the minimum and Sprite #n Sequence 3 Frame Number in REG[1xxxh +1Eh] bits 15-8 as the maximum. Note: This setting must not be used for sprites whose operations are triggered manually using REG[1704h] bits 7-4 and bit 0.
11b	Reserved

10.4.9 2D BitBLT Registers

REG[1800h] BitBLT Control Register 0							
Default = 0000h							Read/Write
n/a							
15	14	13	12	11	10	9	8
n/a							BitBLT Control
7	6	5	4	3	2	1	0

bit 0

BitBLT Control
This bit controls the operation being processed by the 2D BitBLT engine and is used to start a new BitBLT operation. Once the BitBLT operation completes, this bit is automatically cleared to 0b.
For Reads:
When this bit = 0b, no BitBLT operation is being processed.
When this bit = 1b, a BitBLT operation is being processed.
For Writes:
Writing a 0b to this bit has no hardware effect.
Writing a 1b to this bit starts the BitBLT operation.

REG[1802h] BitBLT Control Register 1							
Default = 0000h							Write Only
n/a							
15	14	13	12	11	10	9	8
n/a							BitBLT Software Reset
7	6	5	4	3	2	1	0

bit 0

BitBLT Software Reset (Write Only)
This bit performs a software reset of the 2D BitBLT engine and all BitBLT registers, REG[1800h] ~ REG[1AFEh]. This bit is automatically reset to 0b when the software reset completes.
Writing a 0b to this bit has no hardware effect.
Writing a 1b to this bit initiates a software reset of the 2D BitBLT engine.

REG[1804h] BitBLT Control Register 2
Default = 0000h Read/Write

n/a							
15	14	13	12	11	10	9	8
n/a				BitBLT Color Format Select bits 1-0		Destination BitBLT Memory Mode Select	Source BitBLT Memory Mode Select
7	6	5	4	3	2	1	0

bits 3-2 BitBLT Color Format Select bits [1:0]
These bits select the color format for the 2D BitBLT operation.

Table 10-53 : BitBLT Color Format Selection

REG[1804h] bits 3-2	Color Format
00b	8 bpp
01b	16 bpp
10b	32 bpp
11b	Reserved

Note

The use of bits may be affected when any of the BitBLT functions controlled by the BitBLT Command register (REG[1808h]) are enabled for a specific operation. If Color Expansion is enabled (REG[1808h] bit 5 = 1b), these bits must be set to either 00b (8 bpp) or 01b (16 bpp). If Alpha Combine is enabled (REG[1808h] bit 14 = 1b), the Alpha Map is used and these bits have no effect. When Alpha Blending is enabled (REG[1808h] bit 15 = 1b), the Alpha Blending Source Format bits (REG[1870h] bits 1-0) are used and these bits have no effect.

bit 1 Destination BitBLT Memory Mode Select
This bit determines how the BitBLT destination data is stored in memory. The BitBLT Destination Start Address is specified by the BitBLT Destination Base Address (REG[1818h] ~ REG[181Ah]), BitBLT Destination X Start Position (REG[181Ch]), and BitBLT Destination Y Start Position (REG[181Eh]).
When this bit = 0b, the BitBLT destination data is stored as a rectangular region of memory. When this memory mode is selected, the BitBLT Memory Address Offset register (REG[1824h]) must be programmed with the memory address offset which specifies the offset from the start of one line to the next line.
When this bit = 1b, the BitBLT destination data is stored as a contiguous linear block of memory.

Note

1. When a Read BitBLT operation is selected (REG[1808h] bits 2-0 = 001b), this bit must be defined and the Source BitBLT Memory Mode Select bit (REG[1804h] bit 0) is not required.
2. When a Pattern Fill BitBLT operation is selected (REG[1808h] bits 2-0 = 100b), this bit is ignored.

bit 0 Source BitBLT Memory Mode Select
This bit determines how the BitBLT source data is stored in memory. The BitBLT Source Start Address is specified by the BitBLT Source Base Address (REG[1810h] ~ REG[1812h]), BitBLT Source X Start Position (REG[1814h]), and BitBLT Source Y Start Position (REG[1816h]).
When this bit = 0b, the BitBLT source data is stored as a rectangular region of memory. When this memory mode is selected, the BitBLT Memory Address Offset register (REG[1824h]) must be programmed with the memory address offset which specifies the offset from the start of one line to the next line.
When this bit = 1b, the BitBLT source data is stored as a contiguous linear block of memory.

Note

1. When a Move BitBLT operation is selected (REG[1808h] bits 2-0 = 010b), both the Destination BitBLT Memory Mode Select bit and this bit must be programmed with the appropriate memory mode.
2. When a Write, Read, Solid Fill, or Pattern Fill BitBLT operation is selected (REG[1808h] bits 2-0 = 000b, 001b, 011b, or 100b), this bit is ignored.

REG[1806h] is Reserved

This register is Reserved and should not be written.

REG[1808h] BitBLT Command Register							
Default = 0000h							Read/Write
Alpha Blending Enable	Alpha Combine Enable	Clipping Enable	Reserved				
15	14	13	12	11	10	9	8
ROP Enable	Reverse Direction Enable	Color Expansion Enable	Transparent Enable	n/a	BitBLT Operation bits 2-0		
7	6	5	4	3	2	1	0

bit 15 Alpha Blending Enable
This bit controls the alpha blending with Alpha Map function. Alpha blending can only be enabled when the Move BitBLT operation is selected, REG[1808h] bits 2-0 = 010b.
When this bit = 0b, the alpha blending function is disabled.
When this bit = 1b, the alpha blending function is enabled.

bit 14 Alpha Combine Enable
This bit controls the alpha combine function. Alpha combine can only be enabled when the Write BitBLT operation is selected, REG[1808h] bits 2-0 = 000h.
When this bit = 0b, the alpha combine function is disabled.
When this bit = 1b, the alpha combine function is enabled.

bit 13	Clipping Enable This bit controls the clipping function. Clipping can only be enabled when the Move BitBLT operation is selected, REG[1808h] bits 2-0 = 010h. For further information on clipping, refer to Section 15.2, “BitBLT Terms and Definition” on page 399. When this bit = 0b, the clipping function is disabled. When this bit = 1b, the clipping function is enabled.

Note

When the clipping function is enabled, both the Destination BitBLT Memory Mode Select bit (REG[1804h] bit 1) and the Source BitBLT Memory Mode Select bit (REG[1804h] bit 0) must be set to 0b to specify rectangular memory mode.

bits 12-8	Reserved The default value for these bits is 0_0000b.
bit 7	ROP Enable This bit controls the ROP (Raster Operation) function. ROP can only be enabled when the the Move BitBLT operation is selected, REG[1808h] bits 2-0 = 010b. The ROP to be applied is selected using the ROP Code bits, REG[180Ah] bits 7-0. The address of the ROP pattern is defined by the BitBLT Pattern Start Address bits, REG[1820h] ~ REG[1822h]. When this bit = 0b, the ROP function is disabled. When this bit = 1b, the ROP function is enabled.
bit 6	Reverse Direction Enable This bit controls the reverse direction function. Reverse direction can only be enabled when the Move BitBLT operation is selected, REG[1808h] bits 2-0 = 010b. When this bit = 0b, the reverse direction function is disabled. When this bit = 1b, the reverse direction function is enabled.

Note

When the reverse direction function is enabled, both the Destination BitBLT Memory Mode Select bit (REG[1804h] bit 1) and the Source BitBLT Memory Mode Select bit (REG[1804h] bit 0) must be set for the same memory mode (0,0 or 1,1). The combinations of 1,0 or 0,1 are not supported.

bit 5	Color Expansion Enable This bit controls the color expansion function. Color expansion can only be enabled when the Write or Move BitBLT operation is selected, REG[1808h] bits 2-0 = 000b or 010b. When this bit = 0b, the color expansion function is disabled. When this bit = 1b, the color expansion function is enabled.

Note

For a write BitBLT with color expansion at 1 bpp and a width of 8, only bits 15-8 are used although a 16-bit write is required to the data port. If the width is more than 16, then an extra write to the data port is required in order to the write the extra data.

bit 4 Transparent Enable
This bit controls the transparent function which can be enabled only when the Write or Move BitBLT operation is selected, REG[1808h] bits 2-0 = 000b or 010b.
When this bit = 0b, the transparent function is disabled.
When this bit = 1b, the transparent function is enabled.

bits 2-0 BitBLT Operation bits [2:0]
These bits specify the 2D BitBLT operation to be performed. The Write BitBLT and Move BitBLT operations can be combined with additional BitBLT functions as shown in the following table. For a description of each function, refer to the bit descriptions for REG[1808h] bits 15-4.

Table 10-54 : BitBLT Operation Summary

REG[1808h] bits 2-0	BitBLT Operation	Alpha Blending	Clipping	ROP	Transparent	Alpha Combine	Color Expansion	Reverse Direction
000b	Write[1,6]	—	—	—	√[2]	√	√	—
001b	Read[3,6]	—	—	—	—	—	—	—
010b	Move[4,5]	√	√	√	√	—	√	√
011b	Solid Fill[6]	—	—	—	—	—	—	—
100b	Pattern Fill with transparency[5,6]	—	—	—	—	—	—	—
101b - 111b	Reserved	—	—	—	—	—	—	—

Note

1. When a Write BitBLT operation is selected, only one BitBLT function can be enabled at a time, except for the following combinations:
   - Color Expansion and Transparent functions can be used together for 8 bpp only.
2. The Write BitBLT operation with Transparent function supports only 8 and 16 bpp.
3. When a Read BitBLT operation is selected, the BitBLT Destination Base Address (REG[1818h] ~ REG[181Ah]), BitBLT Destination X Start Position (REG[1814h]), and BitBLT Destination Y Start Position (REG[1816h]) are used instead of the BitBLT source registers.
4. When a Move BitBLT operation is selected, only one BitBLT function can be enabled at a time, except for the following combinations:
   - Color Expansion and Transparent functions can be used together for 8 bpp only
   - ROP and Reverse Direction functions can be used together.
5. A BitBLT software reset (REG[1802h] bit 0 = 1b) must be issued before each Pattern Fill BitBLT operation.
6. For 8 bpp operations, the BitBLT must begin on an even boundary and the width must be an even number of pixels.
7. When Move BitBLT with color expansion is selected the minimum BitBLT width is eight pixels.

REG[180Ah] BitBLT Raster Operation Code Register
Default = 0000h Read/Write

n/a							
15	14	13	12	11	10	9	8
ROP Code bits 7-0							
7	6	5	4	3	2	1	0

bits 7-0 ROP Code bits [7:0]
These bits specify the ROP (Raster Operation) code that defines a logical operation applied to the BitBLT operation. The resulting pixel color is defined as follows.

Pc = { ROP_Code[7] & (Cpat & Csrc & Cdst) }
| { ROP_Code[6] & (Cpat & Csrc & ~Cdst) }
| { ROP_Code[5] & (Cpat & ~Csrc & Cdst) }
| { ROP_Code[4] & (Cpat & ~Csrc & ~Cdst) }
| { ROP_Code[3] & (~Cpat & Csrc & Cdst) }
| { ROP_Code[2] & (~Cpat & Csrc & ~Cdst) }
| { ROP_Code[1] & (~Cpat & ~Csrc & Cdst) }
| { ROP_Code[0] & (~Cpat & ~Csrc & ~Cdst) }

Where:
Pc = Pixel Color of ROP result
ROP_Code = Logical operation to Pat, Src, and Dst
Cpat = Pixel Color of Pattern
Csrc = Pixel Color of Source
Cdst = Pixel Color of Destination

REG[1810h] BitBLT Source Base Address Register 0
Default = 0000h Read/Write

BitBLT Source Base Address bits 15-8							
15	14	13	12	11	10	9	8
BitBLT Source Base Address bits 7-0							
7	6	5	4	3	2	1	0

REG[1812h] BitBLT Source Base Address Register 1
Default = 0000h Read/Write

n/a					BitBLT Source Base Address bits 26-24		
15	14	13	12	11	10	9	8
BitBLT Source Base Address bits 23-16							
7	6	5	4	3	2	1	0

REG[1812h] bits 10-0
REG[1810h] bits 15-0 BitBLT Source Base Address bits [26:0]

These bits specify the base address of the top left corner of the BitBLT source area used for the BitBLT operation, in bytes. These bits are used in conjunction with the BitBLT Source X Start Position (REG[1814h]) and BitBLT Source Y Start Position (REG[1816h]) to determine the actual BitBLT Source Start Address using the following calculation.

BitBLT Source Start Address
= Source Base Address + (X_{src} x (Byte/Pixel)) + (Y_{src} x Address Offset)
= REG[1812h], REG[1810h] + (REG[1814h] x (bytes/pixel)) + (REG[1816h] x REG[1824h])

For further information on the BitBLT Source Start Address, refer to Section 15.2, “BitBLT Terms and Definition” on page 399.

Note

For 8 bpp operations, the BitBLT must begin on an even boundary and the width must be an even number of pixels.

REG[1814h] BitBLT Source X Start Position Register Default = 0000h							Read/Write
n/a				BitBLT Source X Start Position bits 11-8			
15	14	13	12	11	10	9	8
BitBLT Source X Start Position bits 7-0							
7	6	5	4	3	2	1	0

bits 11-0

BitBLT Source X Start Position bits [11:0]
These bits form a 2's complement value that specifies the X start position from the top left corner of the BitBLT source area where the BitBLT source window begins. These bits are used in conjunction with the BitBLT Source Base Address (REG[1810h] ~ REG[1812h]) and BitBLT Source Y Start Position (REG[1816h]) to determine the actual BitBLT Source Start Address using the following calculation.

BitBLT Source Start Address
= Source Base Address + (X_{src} x (Byte/Pixel)) + (Y_{src} x Address Offset)
= REG[1812h], REG[1810h] + (REG[1814h] x (bytes/pixel)) + (REG[1816h] x REG[1824h])

The BitBLT Source X Start Position must be set within the following range.
$-1280 \leq$ BitBLT source X start position ≤ 1280

For further information on the BitBLT Source X Start Position, refer to Section 15.2, "BitBLT Terms and Definition" on page 399.

Note

1. These bits are typically used only for Move BitBLT operations with Clipping enabled, REG[1808h] = 2002h.
2. For 8 bpp operations, the BitBLT must begin on an even boundary and the width must be an even number of pixels.

REG[1816h] BitBLT Source Y Start Position Register Default = 0000h							Read/Write
n/a				BitBLT Source Y Start Position bits 11-8			
15	14	13	12	11	10	9	8
BitBLT Source Y Start Position bits 7-0							
7	6	5	4	3	2	1	0

bits 11-0

BitBLT Source Y Start Position bits [11:0]
These bits form a 2’s complement value that specifies the Y start position from the top left corner of the BitBLT source area where the BitBLT source window begins. These bits are used in conjunction with the BitBLT Source Base Address (REG[1810h] ~ REG[1812h]) and BitBLT Source X Start Position (REG[1814h]) to determine the actual BitBLT Source Start Address using the following calculation.

BitBLT Source Start Address
= Source Base Address + (X_{src} x (Byte/Pixel)) + (Y_{src} x Address Offset)
= REG[1812h], REG[1810h] + (REG[1814h] x (bytes/pixel)) + (REG[1816h] x REG[1824h])

The BitBLT Source Y Start Position must be set within the following range.
$-1024 \leq$ BitBLT source Y start position ≤ 1024

For further information on the BitBLT Source Y Start Position, refer to Section 15.2, “BitBLT Terms and Definition” on page 399.

Note

1. These bits are typically used only for Move BitBLT operations with Clipping enabled, REG[1808h] = 2002h.
2. For 8 bpp operations, the BitBLT must begin on an even boundary and the width must be an even number of pixels.

REG[1818h] BitBLT Destination Base Address Register 0							
Default = 0000h							Read/Write
BitBLT Destination Base Address bits 15-8							
15	14	13	12	11	10	9	8
BitBLT Destination Base Address bits 7-0							
7	6	5	4	3	2	1	0

REG[181Ah] BitBLT Destination Base Address Register 1							
Default = 0000h							Read/Write
n/a					BitBLT Destination Base Address bits 26-24		
15	14	13	12	11	10	9	8
BitBLT Destination Base Address bits 23-16							
7	6	5	4	3	2	1	0

REG[181Ah] bits 10-0
REG[1818h] bits 15-0 BitBLT Destination Base Address bits [26:0]

These bits specify the base address of the top left corner of the BitBLT destination area used for the BitBLT operation, in bytes. These bits are used in conjunction with the BitBLT Destination X Position (REG[181Ch]) and BitBLT Destination Y Position (REG[181Eh]) to determine the actual BitBLT Destination Start Address using the following calculation.

BitBLT Destination Start Address
= Destination Base Address + (X_{dst} x (Byte/Pixel)) + (Y_{dst} x Address Offset)
= REG[181Ah], REG[1818h] + (REG[181Ch] x (bytes/pixel)) + (REG[181Eh] x REG[1824h])

For further information on the BitBLT Destination Start Address, refer to Section 15.2, “BitBLT Terms and Definition” on page 399.

Note

For 8 bpp operations, the BitBLT must begin on an even boundary and the width must be an even number of pixels.

REG[181Ch] BitBLT Destination X Start Position Register Default = 0000h							Read/Write
n/a				BitBLT Destination X Start Position bits 11-8			
15	14	13	12	11	10	9	8
BitBLT Destination X Start Position bits 7-0							
7	6	5	4	3	2	1	0

bits 11-0 BitBLT Destination X Start Position bits [11:0]
These bits form a 2’s complement value that specifies the X start position from the top left corner of the BitBLT destination area where the BitBLT destination window begins. These bits are used in conjunction with the BitBLT Destination Base Address (REG[1818h] ~ REG[181Ah]) and BitBLT Destination Y Start Position (REG[181Eh]) to determine the actual BitBLT Destination Start Address using the following calculation.

BitBLT Destination Start Address
= Destination Base Address + (X_{dst} x (Byte/Pixel)) + (Y_{dst} x Address Offset)
= REG[181Ah], REG[1818h] + (REG[181Ch] x (bytes/pixel)) + (REG[181Eh] x REG[1824h])

The BitBLT Destination X Start Position must be set within the following range.
$-1280 \leq$ BitBLT destination X start position ≤ 1280

For further information on the BitBLT Destination X Start Position, refer to Section 15.2, “BitBLT Terms and Definition” on page 399.

Note

1. These bits are typically used only for Move BitBLT operations with Clipping enabled, REG[1808h] = 2002h.
2. For 8 bpp operations, the BitBLT must begin on an even boundary and the width must be an even number of pixels.

REG[181Eh] BitBLT Destination Y Start Position Register
Default = 0000h — Read/Write

n/a				BitBLT Destination Y Start Position bits 11-8			
15	14	13	12	11	10	9	8
BitBLT Destination Y Start Position bits 7-0							
7	6	5	4	3	2	1	0

bits 11-0 BitBLT Destination Y Start Position bits [11:0]
These bits form a 2’s complement value that specifies the Y start position from the top left corner of the BitBLT destination area where the BitBLT destination window begins. These bits are used in conjunction with the BitBLT Destination Base Address (REG[1818h] ~ REG[181Ah]) and BitBLT Destination Y Start Position (REG[181Eh]) to determine the actual BitBLT Destination Start Address using the following calculation.

BitBLT Destination Start Address
= Destination Base Address + (X_{dst} x (Byte/Pixel)) + (Y_{dst} x Address Offset)
= REG[181Ah], REG[1818h] + (REG[181Ch] x (bytes/pixel)) + (REG[181Eh] x REG[1824h])

The BitBLT Destination Y Start Position must be set within the following range.
$-1024 \leq$ BitBLT destination Y start position ≤ 1024

For further information on the BitBLT Destination Y Start Position, refer to Section 15.2, “BitBLT Terms and Definition” on page 399.

Note

1. These bits are typically used only for Move BitBLT operations with Clipping enabled, REG[1808h] = 2002h.
2. For 8 bpp operations, the BitBLT must begin on an even boundary and the width must be an even number of pixels.

REG[1820h] BitBLT Pattern Start Address Register 0 Default = 0000h							Read/Write
BitBLT Pattern Start Address bits 15-8							
15	14	13	12	11	10	9	8
BitBLT Pattern Start Address bits 7-0							
7	6	5	4	3	2	1	0

REG[1822h] BitBLT Pattern Start Address Register 1 Default = 0000h							Read/Write
n/a					BitBLT Pattern Start Address bits 26-24		
15	14	13	12	11	10	9	8
BitBLT Pattern Start Address bits 23-16							
7	6	5	4	3	2	1	0

REG[1822h] bits 10-0
REG[1820h] bits 15-0 BitBLT Pattern Start Address bits [26:0]

When the BitBLT Operation bits are set for Pattern Fill with Transparency (REG[1808h] bits 2-0 = 100b), these bits are used to specify the Pattern Base Address, Pattern Line Offset, and Pixel Offset. Each value is determined from a portion of this register field and differs based on the selected color format, as shown in the following table. These bits are also used to define the address of the ROP pattern when ROP is enabled, REG[1808h] bit 7 = 1b.

Table 10-55 : BitBLT Pattern Start Address Summary

Color Format	Pattern Base[26:0]	Pattern Line Offset[2:0]	Pixel Offset[4:0]
8 bpp	Pattern Start Address[26:7] + 000_0000b	Pattern Start Address[5:3]	00b + Pattern Start Address[2:0]
16 bpp	Pattern Start Address[26:8] + 0000_0000b	Pattern Start Address[6:4]	0b + Pattern Start Address[3:1] + 0b
32 bpp	Pattern Start Address[26:9] + 0_0000_0000b	Pattern Start Address[7:5]	Pattern Start Address[4:2] + 00b

REG[1824h] BitBLT Memory Address Offset Register Default = 0000h							Read/Write
n/a			BitBLT Memory Address Offset bits 12-8				
15	14	13	12	11	10	9	8
BitBLT Memory Address Offset bits 7-0							
7	6	5	4	3	2	1	0

bits 12-0 BitBLT Memory Address Offset bits [12:0]

These bits specify the memory address offset for the Source and Destination BitBLT areas used for BitBLT operations, from the starting word of line *n* to the starting word of line *n + 1*. These bits are only used for address calculation when the BitBLT operation is configured for a rectangular region of memory (see REG[1804h] bits 1-0).

REG[1826h] BitBLT Width Register
Default = 0001h Read/Write

n/a					BitBLT Width bits 10-8		
15	14	13	12	11	10	9	8
BitBLT Width bits 7-0							
7	6	5	4	3	2	1	0

bits 10-0

BitBLT Width bits [10:0]
These bits determine the width of the BitBLT operation, in pixels. These bits apply to both the source and destination BitBLT windows. The BitBLT width must be within the following range.

1 ≤ BitBLT width ≤ 1280

Note

1. If 0000h is written to these bits, the register will automatically be set to 0001h.
2. For 8 bpp operations, the BitBLT must begin on an even boundary and the width must be an even number of pixels.
3. When Move BitBLT with Color Expand is selected, REG[1808h] bits 2-0 = 010b AND REG[1808h] bit 5 = 1b, the minimum width is 8 pixels.

REG[1828h] BitBLT Height Register
Default = 0001h Read/Write

n/a					BitBLT Height bits 10-8		
15	14	13	12	11	10	9	8
BitBLT Height bits 7-0							
7	6	5	4	3	2	1	0

bits 10-0

BitBLT Height bits [10:0]
These bits specify the height of the BitBLT operation, in lines. These bits apply to both the source and destination BitBLT windows. The BitBLT height must be set within the following range.

1 ≤ BitBLT height ≤ 1024

Note

If a 0000h is written to these bits, the register will automatically be set to 0001h.

REG[1834h] BitBLT Clipping X Start Position Register
Default = 0000h Read/Write

n/a					BitBLT Clipping X Start Position bits 10-8		
15	14	13	12	11	10	9	8
BitBLT Clipping X Start Position bits 7-0							
7	6	5	4	3	2	1	0

bits 10-0

BitBLT Clipping X Start Position bits [10:0]
These bits specify the X start position from the BitBLT Destination Base Address (REG[1818h] ~ REG[181Ah]) where the BitBLT clipping area begins. These bits are used in conjunction with the BitBLT Destination Base Address (REG[1818h] ~ REG[181Ah]) and BitBLT Clipping Y Start Position (REG[1836h]) to determine the actual BitBLT Clipping Start Address using the following calculation.

BitBLT Clipping Start Address
= Destination Base Address + (X_{clp} x (Bytes per Pixel)) + (Y_{clp} x Address Offset)
= REG[1818h], REG[181Ah] + (REG[1834h] x (bytes per pixel)) + (REG[1836h] x REG[1824h])

The BitBLT Clipping X Start Position must be set within the following range.
$0 \leq$ BitBLT width ≤ 1279

For further information on the BitBLT Clipping X Start Position, refer to Section 15.2, “BitBLT Terms and Definition” on page 399.

Note

The following formula must be valid when programming REG[1834h] bits 10-0.
REG[1834h] bits 10-0 + REG[1838h] bits 10-0 < ((REG[1824h] bits 12-0) ÷ bytes per pixel)-1

REG[1836h] BitBLT Clipping Y Start Position Register
Default = 0000h Read/Write

n/a						BitBLT Clipping Y Start Position bits 9-8	
15	14	13	12	11	10	9	8
BitBLT Clipping Y Start Position bits 7-0							
7	6	5	4	3	2	1	0

bits 9-0 BitBLT Clipping Y Start Position bits [9:0]
These bits specify the Y start position from the BitBLT Destination Base Address (REG[1818h] ~ REG[181Ah]) where the BitBLT clipping area begins. These bits are used in conjunction with the BitBLT Destination Base Address (REG[1818h] ~ REG[181Ah]) and BitBLT Clipping X Start Position (REG[1834h]) to determine the actual BitBLT Clipping Start Address using the following calculation.

BitBLT Clipping Start Address
= Destination Base Address + (X_{clp} x (Bytes per Pixel)) + (Y_{clp} x Address Offset)
= REG[1818h], REG[181Ah] + (REG[1834h] x (bytes per pixel)) + (REG[1836h] x REG[1824h])

The BitBLT Clipping Y Start Position must be set within the following range.
$0 \leq$ BitBLT width ≤ 1023

For further information on the BitBLT Clipping Y Start Position, refer to Section 15.2, “BitBLT Terms and Definition” on page 399.

REG[1838h] BitBLT Clipping Width Register
Default = 0001h Read/Write

n/a					BitBLT Clipping Width bits 10-8		
15	14	13	12	11	10	9	8
BitBLT Clipping Width bits 7-0							
7	6	5	4	3	2	1	0

bits 10-0 BitBLT Clipping Width bits [10:0]
These bits determine the width of the BitBLT clipping area, in pixels. The BitBLT clipping area width must be set within the following range.
$1 \leq$ BitBLT clipping width ≤ 1280

Note
1. If 0000h is written to these bits, the register will automatically be set to 0001h.
2. The following formula must be valid when programming REG[1838h] bits 10-0.
REG[1834h] bits 10-0 + REG[1838h] bits 10-0 < ((REG[1824h] bits 12-0) ÷ bytes per pixel) - 1

REG[1840h] BitBLT Clipping Height Register
Default = 0001h Read/Write

n/a					BitBLT Clipping Height bits 10-8		
15	14	13	12	11	10	9	8
BitBLT Clipping Height bits 7-0							
7	6	5	4	3	2	1	0

bits 10-0 BitBLT Clipping Height bits [10:0]
These bits determine the height of the BitBLT clipping area, in lines. The BitBLT clipping area height must be set within the following range.
$1 \leq$ BitBLT clipping height ≤ 1024

Note
If 0000h is written to these bits, the register will automatically be set to 0001h.

REG[1842h] BitBLT Clipping Status Register
Default = 0000h Read Only

n/a							
15	14	13	12	11	10	9	8
n/a						Reserved	BitBLT Clipping Out Of Range
7	6	5	4	3	2	1	0

bit 1 Reserved
The default value for this bit is 0b.

bit 0 BitBLT Clipping Out Of Range (Read Only)
This bit indicates whether a BitBLT window is within the clipping area as defined by REG[1818h] ~ REG[181Ah] and REG[1834h] ~ REG[1840h].
When this bit = 0b, a BitBLT window or portion of a BitBLT window is within the specified clipping area.
When this bit = 1b, all BitBLT windows are totally out of the clipping area.

To clear this bit, start a new BitBLT operation by writing a 1b to the BitBLT Control bit, REG[1800h] bit 0.

REG[1850h] BitBLT Background Color Register 0							
Default = 0000h							Read/Write
BitBLT Background Color bits 15-8							
15	14	13	12	11	10	9	8
BitBLT Background Color bits 7-0							
7	6	5	4	3	2	1	0

REG[1852h] BitBLT Background Color Register 1							
Default = 0000h							Read/Write
BitBLT Background Color bits 31-24							
15	14	13	12	11	10	9	8
BitBLT Background Color bits 23-16							
7	6	5	4	3	2	1	0

REG[1852h] bits 15-0
REG[1850h] bits 15-0 BitBLT Background Color bits [31:0]

These bits specify the background color which is used as the key color for transparent BitBLT operations. For each color depth, a different number of bits are used.
For a color depth of 32 bpp (REG[1804h] bits 3-2 = 10b), all of bits 31-0 are used.
For a color depth of 16 bpp (REG[1804h] bits 3-2 = 01b), only bits 15-0 are used.
For a color depth of 8 bpp (REG[1804h] bits 3-2 = 10b), only bits 7-0 are used.

REG[1854h] BitBLT Foreground Color Register 0							
Default = 0000h							Read/Write
BitBLT Foreground Color bits 15-8							
15	14	13	12	11	10	9	8
BitBLT Foreground Color bits 7-0							
7	6	5	4	3	2	1	0

REG[1856h] BitBLT Foreground Color Register 1							
Default = 0000h							Read/Write
BitBLT Foreground Color bits 31-24							
15	14	13	12	11	10	9	8
BitBLT Foreground Color bits 23-16							
7	6	5	4	3	2	1	0

REG[1856h] bits 15-0
REG[1854h] bits 15-0 BitBLT Foreground Color bits [31:0]

These bits specify the foreground color which is used for Solid Fill BitBLT operations, REG[8008h] bits 2-0 = 011b. For each color depth a different number of bits are used.
For a color depth of 32 bpp (REG[1804h] bits 3-2 = 10b), all of bits 31-0 are used.
For a color depth of 16 bpp (REG[1804h] bits 3-2 = 01b), only bits 15-0 are used.
For a color depth of 8 bpp (REG[1804h] bits 3-2 = 10b), only bits 7-0 are used.

REG[1860h] BitBLT Color Expansion Start Position Register Default = 0000h							Read/Write
n/a							
15	14	13	12	11	10	9	8
n/a					Color Expansion Start Bit Position bits 2-0		
7	6	5	4	3	2	1	0

bits 2-0

Color Expansion Start Bit Position bits [2:0]
These bits specify the start bit position when 1-bit Color Expansion (REG[1862h] bits 1-0 = 00b) is enabled for a Write or Move BitBLT operation, REG[1808h] bit 5 = 1b. For 2/4/8-bit color expansion, these bits should be set to 000b.

Note

For color expansion, the Start Position is specified by SrcStartAddr[0] + Color Expansion Start Bit Position (i.e. REG[1810h] bit 0 + REG[1860h] bit 2-0). This applies for both the Move BitBLT and Write BitBLT.

REG[1862h] BitBLT Color Expansion Bit Format Register
Default = 0000h Read/Write

n/a							
15	14	13	12	11	10	9	8
n/a						Source Bit Format bits 1-0	
7	6	5	4	3	2	1	0

bits 1-0 Source Bit Format bits [1:0]
These bits specify the bit format of the source for Color Expansion operations. Color expansion is performed from MSB to LSB using the source as indexes into a LUT (Look Up Table). However, for 1-bit source bit format, the LUT index used for color expansion is byte swapped. Also, for 2/4/8-bit source bit formats, the LUT index used for color expansion is reversed (lsb to msb within the source bitfield). For example, for a 2-bit source, data of 01b actually color expands from LUT index 2 instead of LUT index 1.

Table 10-56 : Source Bit Format Selection

REG[1862h] bits 1-0	Source Bit Format	Example Data = 1234h (0001 0010 0011 0100b)
00b	1-bit	Color Expansion uses the following LUT indexes: 0h 0h 1h 1h 0h 1h 0h 0h 0h 0h 0h 1h 0h 0h 1h 0h
01b	2-bit	Translated Data: 00 10 00 01 00 11 10 00 LUT Index: 0h 2h 0h 1h 0h 3h 2h 0h
10b	4-bit	Translated Data: 1000 0100 1100 0010 LUT Index: 8h 4h Ch 2h
11b	8-bit	Translated Data: 0100 1000 0010 1100 LUT Index: 48h 2Ch

Note

For further information on the LUT structure, refer to the bit description for REG[1900h] ~ REG[1AFEh].

REG[1870h] BitBLT Alpha Blending Source Format Register
Default = 0000h Read/Write

n/a							
15	14	13	12	11	10	9	8
n/a						Alpha Blending Source Format bits 1-0	
7	6	5	4	3	2	1	0

bits 1-0 Alpha Blending Source Format bits [1:0]
These bits specify the format of the source data used when Alpha Blending is enabled, REG[1808h] bit 15 = 1b. When alpha blending is enabled, these bits are used and the BitBLT Color Format Select bits (REG[1804h] bits 3-2) have no effect.

Table 10-57 : Alpha Blending Source Format

REG[1870h] bits 1-0	Source Format	Destination	Result (Destination)
00b	ARGB 1:5:5:5	RGB 5:6:5	RGB 5:6:5
01b	ARGB 4:4:4:4	RGB 5:6:5	RGB 5:6:5
10b	ARGB 8:8:8:8	RGB 5:6:5	RGB 5:6:5
11b	Reserved	Reserved	Reserved

REG[1872h] BitBLT Constant Alpha Register
Default = 0000h Read/Write

n/a							
15	14	13	12	11	10	9	8
Constant Alpha Value bits 7-0							
7	6	5	4	3	2	1	0

bits 7-0 Constant Alpha Value bits [7:0]
These bits specify the 8-bit constant alpha value. When alpha blending is enabled (REG[1808h] bit 15 = 1b), these bits a used in conjunction with the Destination Alpha Value bits (REG[1874h] bits 9-8) and the Source Alpha Value bits (REG[1874h] bits 1-0).

Note

Bit 0 has no effect with Alpha Blending.

REG[1874h] BitBLT Alpha Value Selection Register
Default = 0001h Read/Write

n/a						Destination Alpha Value bits 1-0	
15	14	13	12	11	10	9	8
n/a						Source Alpha Value bits 1-0	
7	6	5	4	3	2	1	0

The output color is calculated using the following formula.
OutColor (RGB)
= Source Alpha x Source Color(RGB) + Destination Alpha x Destination Color(RGB)

There are many possible combinations of Destination and Source Alpha Value settings. The following table summarizes the most useful or recommended settings.

Table 10-58 : Destination and Source Alpha Value Recommended Settings

REG[1874h] bits 9-8 and 1-0	Description
0100b	Source = Alpha, Destination = (1-Alpha)
0001b	Source = (1-Alpha), Destination = Alpha
1110b	Source = Constant Alpha, Destination = (1-Constant Alpha)
1011b	Source = (1-Constant Alpha), Destination = Constant Alpha

bits 9-8 Destination Alpha Value bits [1:0]
These bits specify the destination alpha value which is used when alpha blending is enabled, REG[1808h] bit 15 = 1b.

Table 10-59 : Destination Alpha Selection

REG[1874h] bits 9-8	Destination Alpha
00b	Alpha (in Alpha Map)
01b	1 - Alpha
10b	Constant-Alpha
11b	1 - Constant-Alpha

bits 1-0 Source Alpha Value bits [1:0]
These bits specify the source alpha value which is used when alpha blending is enabled, REG[1808h] bit 15 = 1b.

Table 10-60 : Source Alpha Selection

REG[1874h] bits 1-0	Source Alpha
00b	Alpha (in Alpha Map)
01b	1 - Alpha
10b	Constant-Alpha
11b	1 - Constant-Alpha

REG[1876h] BitBLT Alpha Combine Alpha Map Register
Default = 0000h Read/Write

n/a							
15	14	13	12	11	10	9	8
n/a						Alpha Map bits 1-0	
7	6	5	4	3	2	1	0

bits 1-0 Alpha Map bits [1:0]
These bits select the alpha value type format used when the alpha combine function is enabled, REG[1808h] bit 14 = 1b. When using alpha combine, the non-alpha RGB Map is replaced with the new alpha RGB map when the new alpha value (1/4/8-bit) is written as Source from the Host CPU. The alpha value written to the BitBLT FIFO Data Port (REG[1896h] bits 15-0) is translated to memory as follows:

Table 10-61 : : Alpha Combine Mapping

Alpha Map	Pixel$_0$ Alpha Bits[msb:lsb]	Pixel$_1$ Alpha Bits[msb:lsb]	Pixel$_{(16/\text{Alpha Map}) - 1}$ Alpha Bits[msb:lsb]
1	BLT FIFO Bit 15	BLT FIFO Bit 14	BLT FIFO Bit 0
4	BLT FIFO Bits [12:15]	BLT FIFO Bits [8:11]	BLT FIFO Bits [0:3]
8	BLT FIFO Bits [8:15]	BLT FIFO Bits[0:7]	BLT FIFO Bits[0:7]

REG[1880h] BitBLT Interrupt Status Register
Default = 0000h Read/Write

n/a							
15	14	13	12	11	10	9	8
n/a							BitBLT Interrupt Status
7	6	5	4	3	2	1	0

bit 0 BitBLT Interrupt Status
This bit indicates the status of the BitBLT interrupt which occurs when the BitBLT transfer completes. This bit is not masked by the BitBLT Enable bit, REG[1882h] bit 0.
When this bit = 0b, a BitBLT interrupt has not occurred.
When this bit = 1b, a BitBLT interrupt has occurred.

To clear this status bit, write a 1b to this bit.

REG[1882h] BitBLT Interrupt Control Register							
Default = 0001h							Read/Write
n/a							
15	14	13	12	11	10	9	8
n/a							BitBLT Interrupt Enable
7	6	5	4	3	2	1	0

bit 0 BitBLT Interrupt Enable
This bit controls whether the BitBLT interrupt causes an interrupt request. A BitBLT interrupt occurs when the BitBLT transfer completes. This bit does not mask the status of the BitBLT interrupt which is indicated by the BitBLT Interrupt Status bit, REG[1880h] bit 0.
When this bit = 0b, a BitBLT interrupt will not cause an interrupt request.
When this bit = 1b, a BitBLT interrupt will cause an interrupt request.

REG[1886h] is Reserved

This register is Reserved and should not be written.

REG[1890h] BitBLT FIFO Status Register 0							
Default = 0001h							Read Only
n/a							
15	14	13	12	11	10	9	8
n/a						BitBLT FIFO Full Flag	BitBLT FIFO Empty Flag
7	6	5	4	3	2	1	0

bit 1 BitBLT FIFO Full Flag (Read Only)
This bit indicates whether the 16 entry BitBLT FIFO is full.
When this bit = 0b, the BitBLT FIFO is not full.
When this bit = 1b, the BitBLT FIFO is full.

bit 0 BitBLT FIFO Empty Flag (Read Only)
This bit indicates whether the 16 entry BitBLT FIFO is empty.
When this bit = 0b, the BitBLT FIFO is not empty.
When this bit = 1b, the BitBLT FIFO is empty.

REG[1892h] BitBLT FIFO Status Register 1							
Default = 0010h							Read Only
n/a							
15	14	13	12	11	10	9	8
n/a			BitBLT FIFO Available Write Entries bits 4-0				
7	6	5	4	3	2	1	0

bits 4-0 BitBLT FIFO Available Write Entries bits [4:0] (Read Only)
These bits indicate the number of 16-bit write entries available in the BitBLT FIFO for CPU write operations. The maximum number of BitBLT FIFO entries is 16.

REG[1894h] BitBLT FIFO Status Register 2

Default = 0000h — Read Only

n/a							
15	14	13	12	11	10	9	8
n/a			BitBLT FIFO Available Read Entries bits 4-0				
7	6	5	4	3	2	1	0

bits 4-0 — BitBLT FIFO Available Read Entries bits [4:0] (Read Only)
These bits indicate the number of 16-bit read entries available in the BitBLT FIFO for CPU read operations. The maximum number of BitBLT FIFO entries is 16.

REG[1896h] BitBLT FIFO Data Port Register

Default = 0000h — Read/Write

BitBLT FIFO Data Port bits 15-8							
15	14	13	12	11	10	9	8
BitBLT FIFO Data Port bits 7-0							
7	6	5	4	3	2	1	0

bits 15-0 — BitBLT FIFO Data Port bits [15:0]
These bits are the data port used to read data from, or write data to the 16 entry BitBLT FIFO. For Write BitBLT operations, appropriate data must be written to the BitBLT FIFO through this port. For Read BitBLT operations, the data can be read from this port.

REG[1900h] ~ REG[1AFEh] BitBLT Color Expansion LUT Data Registers

Default = xxxxh — Read/Write

BitBLT Color Expansion LUT Data bits 15-8							
15	14	13	12	11	10	9	8
BitBLT Color Expansion LUT Data bits 7-0							
7	6	5	4	3	2	1	0

bits 15-0 — BitBLT Color Expansion LUT Data bits [15:0]
These bits define the color expansion data used for BitBLT operations. There are 256 16-bit entries in this Look-Up Table (LUT) from REG[1900h] ~ REG[1AFEh].

Note

These registers must not be changed while a BitBLT operation with the color expansion function enabled is in progress.

10.4.10 Memory Controller Registers

REG[1C00h] Memory Control Register Default = 0000h							Read/Write
n/a							
15	14	13	12	11	10	9	8
SDRAM Write Protect Enable	n/a	Reserved	Reserved	Reserved	Reserved		
7	6	5	4	3	2	1	0

Note

This register must be set to 11h before the SDRAM memory is enabled using REG[1C02h] bit 0.

bit 7 SDRAM Write Protect Enable
This bit determines whether the external SDRAM memory is write protected.
When this bit = 0b, write protection is disabled. (default)
When this bit = 1b, write protection is enabled.

bit 5 Reserved
This bit must be set to 0b.

bit 4 Reserved
This bit must be set to 1b.

bit 3 Reserved
This bit must be set to 0b.

bits 2-0 Reserved
These bits must be set to 001b.

REG[1C02h] Memory Configuration Register 0 Default = 0002h							Read/Write
n/a							
15	14	13	12	11	10	9	8
Memory Initialized (RO)	n/a				SDRAM External Bus Width bits 1-0		Memory Enable
7	6	5	4	3	2	1	0

bit 7 Memory Initialized (Read Only)
This bit indicates the initialization status of the external SDRAM memory. This bit must be checked after the first time the SDRAM memory is enabled using the Memory Enable bit (REG[1C02h] bit 0) to determine if the SDRAM initialization process has completed.
When this bit = 0b, the SDRAM has not be initialized.
When this bit = 1b, the SDRAM has been initialized and can be accessed when REG[1C02h] bit 0 = 1b.

Note

To access the SDRAM, both the Memory Initialized bit and the Memory Enable bit (REG[1C02h] bit 0) must equal 1b.

bits 2-1 SDRAM External Bus Width bits [1:0]
These bits select the bus width for the external SDRAM Memory.

Table 10-62 : SDRAM External Bus Width Selection

REG[1C02h] bits 2-1	SDRAM External Bus Width
00b	Reserved
01b	16-bit (default)
10b	32-bit
11b	Reserved

bit 0 Memory Enable
This bit enables the external SDRAM memory. If the memory has not been initialized, an initialization sequence is sent to the SDRAM when this bit is set to 1b. When this bit is set, the Memory Initialized bit (REG[1C02h] bit 7) must be checked to confirm that initialization is complete **before** accessing the SDRAM memory.
When this bit = 0b, the SDRAM memory is disabled.
When this bit = 1b, the SDRAM memory is enabled and the initialization process is started. The SDRAM can be accessed once REG[1C02h] bit 7 = 1b.

Note

1. To access the SDRAM, both the Memory Initialized bit (REG[1C02h] bit 7) and the Memory Enable bit must equal 1b.
2. To lower power consumption, the SDRAM should be put into self-refresh mode before disabling the SDRAM interface.
3. REG[1C00h] must be set to 11h before enabling the SDRAM memory.

<table>
<tr><td colspan="8"><b>REG[1C04h] Memory Configuration Register 1</b>
Default = 7AAFh Read/Write</td></tr>
<tr><td colspan="4">Active to Precharge Cycle bits 3-0</td><td colspan="4">Auto Refresh to Active Cycle bits 3-0</td></tr>
<tr><td>15</td><td>14</td><td>13</td><td>12</td><td>11</td><td>10</td><td>9</td><td>8</td></tr>
<tr><td colspan="2">Write to Precharge bits 1-0</td><td colspan="2">Return to Precharge bits 1-0</td><td colspan="2">RAS to CAS Delay bits 1-0</td><td colspan="2">CAS Latency bits 1-0</td></tr>
<tr><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
</table>

bits 15-12 Active to Precharge Cycle bits [3:0]
These bits are used to configure the Active to Precharge cycle for the external SDRAM, in MEMCLK periods.

Active to Precharge Cycle = REG[1C04h] bits 15-12 + 1

Note

If the Active to Precharge Cycle is configured to a value less than the number of cycles required for a specific operation, this setting is ignored.

bits 11-8 Auto Refresh to Active Cycle bits [3:0]
These bits control the minimum amount of time, in SDRAM clock cycles, an active command can be issued after an auto refresh command.

Table 10-63 : Auto Refresh to Active Cycle Time

REG[1C04h] bits 11-8	Refresh to Active Time	REG[1C04h] bits 11-8	Refresh to Active Time
0000b	Reserved	1000b	9 SDRAM clocks
0001b	Reserved	1001b	10 SDRAM clocks
0010b	3 SDRAM clocks	1010b (default)	11 SDRAM clocks
0011b	4 SDRAM clocks	1011b	12 SDRAM clocks
0100b	5 SDRAM clocks	1100b	13 SDRAM clocks
0101b	6 SDRAM clocks	1101b	14 SDRAM clocks
0110b	7 SDRAM clocks	1110b	15 SDRAM clocks
0111b	8 SDRAM clocks	1111b	16 SDRAM clocks

bits 7-6 Write to Precharge bits [1:0] (TWR)
These bits are used to configure the Write to Precharge value for the external SDRAM, in cycles.

Table 10-64 : Write to Precharge Selection

REG[1C04h] bits 7-6	Write to Precharge
00b	Reserved
01b	2 Clocks
10b (default)	3 Clocks
11b	4 Clocks

Note

1. The Write to Precharge value only has an effect if:
   RAS to CAS delay + Write to Precharge time + time from first write latched to last write latched > Active to Precharge time (see REG[1C04h] bits 15-12)
2. If the Write to Precharge is set to 2 clocks, the RAS to CAS delay must be set to 3 clocks (REG[1C04h] bits 3-2 = 11b).

bits 5-4 Return to Precharge bits [1:0] (TRP)
These bits are used to configure the Return to Precharge value for the external SDRAM, in cycles.

Table 10-65 : Return to Precharge Selection

REG[1C04h] bits 5-4	Return to Precharge
00b	Reserved
01b	2 Clocks
10b (default)	3 Clocks
11b	4 Clocks

bits 3-2 RAS to CAS Delay bits [1:0] (TRCD)
These bits are used to configure the delay time between RAS# and CAS# for the external SDRAM, in cycles.

Table 10-66 : RAS to CAS Delay Selection

REG[1C04h] bits 3-2	RAS to CAS Delay
00b	Reserved
01b	Reserved
10b	2 Clocks
11b (default)	3 Clocks

bits 1-0 CAS Latency bits [1:0]
These bits are used to configure the CAS Latency for the external SDRAM, in cycles.

Table 10-67 : CAS Latency Selection

REG[1C04h] bits 1-0	CAS Latency
00b	Reserved
01b	Reserved
10b	2 Clocks
11b (default)	3 Clocks

<table>
<tr><td colspan="8"><b>REG[1C06h] Memory Configuration Register 2</b>
Default = C010h Read/Write</td></tr>
<tr><td>Reserved</td><td>Dynamic SDRAM CKE Control</td><td colspan="6">n/a</td></tr>
<tr><td>15</td><td>14</td><td>13</td><td>12</td><td>11</td><td>10</td><td>9</td><td>8</td></tr>
<tr><td colspan="2">n/a</td><td colspan="2">Reserved</td><td>Auto Precharge</td><td colspan="2">Memory Size bits 1-0</td><td>Reserved</td></tr>
<tr><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
</table>

bit 15 Reserved
The default value for this bit is 1b.

bit 14 Dynamic SDRAM CKE Control
This bit controls when the clock enable (MEMCKE) to the external SDRAM memory is asserted.
When this bit = 0b, clock enable is always asserted regardless of the SDRAM bus state.
When this bit = 1b, clock enable to the external SDRAM is driven high or low depending on the state of the SDRAM bus. (default)

bits 5-4 Reserved
The default value for these bits is 01b.

bit 3 Auto Precharge
This bit controls whether the SDRAM memory banks are precharged.
When this bit = 0b, the SDRAM memory banks are not precharged. (default)
When this bit = 1b, the SDRAM memory banks are precharged.

bits 2-1 Memory Size bits [1:0]
These bits configure the size of the external SDRAM memory connected to the S1D13513.

Table 10-68 : Memory Size Selection

REG[1C06h] bits 2-1	Memory Size
00b (default)	128 Mbit
01b	256 Mbit
10b	512 Mbit
11b	64 Mbit

The following table summarizes the address widths for each memory interface configuration.

Table 10-69 : Memory Interface Address Widths

	Memory Size (REG[1C06h] bits 2-1)			
16-bit Memory Interface	**64 Mbit**	**128 Mbit**	**256 Mbit**	**512 Mbit**
Column Address Width	8	9	9	10
Bank Address Width	2	2	2	2
Row Address Width	12	12	13	13

32-bit Memory Interface	64 Mbit	128 Mbit	256 Mbit	512 Mbit
Column Address Width	8	8	9	9
Bank Address Width	2	2	2	2
Row Address Width	11	12	12	13

bit 0 Reserved
The default value for this bit is 0b.

REG[1C08h] Memory Advanced Configuration Register
Default = 0F1Ah Read/Write

Self Refresh Re-entry Cycle Count bits 7-0							
15	14	13	12	11	10	9	8
n/a	Reserved				Reserved	Self Refresh Re-entry Control	Self Refresh Mode Enable (WO)
7	6	5	4	3	2	1	0

bits 15-8 Self Refresh Re-entry Cycle Count bits [7:0]
These bits specify the number of cycles that must take place without further accesses before the SDRAM will re-enter self refresh mode. This happens when the SDRAM has been awoken from self refresh mode for a read or write access **and** the Self Refresh Re-entry Control bit is set to 1b (REG[1C08h] bit 1 = 1b). The default value is 0Fh and the minimum value is 02h. The count clock uses 1/4 of MEMCLK.
Self Refresh Re-entry delay = (REG[1C08h] bits 15-8) x 4 SDRAM clocks

bits 6-3 Reserved
The default value for these bits is 0011b.

bit 2 Reserved
The default value for this bit is 0b.

bit 1 Self Refresh Re-entry Control
This bit controls whether the SDRAM will re-enter self refresh mode after it has been awoken for a read or write access. If this function is enabled, the SDRAM will re-enter self refresh mode if there are no further accesses for the number of cycles specified by the Self Refresh Re-entry Cycle Counter bits, REG[1C08h] bits 15-8.
When this bit = 0b, the SDRAM will not re-enter self refresh mode. (default)
When this bit = 1b, the SDRAM will re-enter self refresh mode if there are no further accesses for the specified number of cycles.

Note
If the SDRAM is brought out of self refresh mode by setting REG[1C08h] bit 0 to 0b, the SDRAM will still re-enter self refresh if this bit is set to 1b.

bit 0 Self Refresh Mode Enable (Write Only)
This bit controls self refresh mode for the SDRAM. When the Self Refresh Re-entry Control bit is set to 1b (REG[1C08h] bit 1 = 1b), the SDRAM will automatically re-enter self refresh mode after the specified number of cycles.
When this bit = 0b, exits the SDRAM from self refresh mode (disabled). (default)
When this bit = 1b, places the SDRAM in self refresh mode (enabled).

Note
The exit self refresh to active time is 3 MEMCLKs. To prevent data corruption, verify that this time does not violate the SDRAM specification before entering self refresh mode.

REG[1C0Ah] Memory Initialization Configuration Register
Default = 0000h Read/Write

n/a							
15	14	13	12	11	10	9	8
n/a				SDRAM Initialization Sequence	SDRAM Load Mode	SDRAM Auto Refresh	SDRAM All Banks Precharge
7	6	5	4	3	2	1	0

Note

This register allows manual triggering of commands to the SDRAM, such as initializing and precharging. For normal use these bits do not have to be changed.

bit 3 SDRAM Initialization Sequence
This bit manually triggers a SDRAM initialization sequence.
When this bit = 0b, a SDRAM initialization sequence is not performed. (default)
When this bit = 1b, a SDRAM initialization sequence is performed. This bit is automatically cleared to 0b after it is written.

Note

1. This bit must not be set to 1b unless the SDRAM interface is idle.
2. If this bit is set to 1b while the SDRAM is in self refresh mode, the command is sent as the next command proceeding the next SDRAM write or read operation.

bit 2 SDRAM Load Mode
This bit manually triggers a load mode register command to the SDRAM. For normal use this bit does not have to be changed.
When this bit = 0b, a load mode register command is not issued. (default)
When this bit = 1b, a load mode register command is issued to the SDRAM. This bit is automatically cleared to 0b after it is written.

Note

1. This bit must not be set to 1b unless the SDRAM interface is idle.
2. If this bit is set to 1b while the SDRAM is in self refresh mode, the command is sent as the next command proceeding the next SDRAM write or read operation.

bit 1 SDRAM Auto Refresh
This bit manually triggers a precharge command followed by an auto refresh command to the SDRAM. For normal use this bit does not have to be changed.
When this bit = 0b, precharge and auto refresh commands are not issued. (default)
When this bit = 1b, precharge and auto refresh commands are issued with, but proceeding, the next SDRAM operation. This bit is automatically cleared to 0b after it is written.

bit 0 SDRAM All Banks Precharge
This bit manually triggers a precharge all banks command to the SDRAM. For normal use this bit does not have to be changed.
When this bit = 0b, a precharge all banks command is not issued. (default)
When this bit = 1b, a precharge all banks command is issued with, but proceeding, the next SDRAM operation. This bit is automatically cleared to 0b after it is written.

REG[1C0Ch] Memory Refresh Timer Register Default = 0411h							Read/Write
SDRAM Auto Refresh Timer bits 15-8							
15	14	13	12	11	10	9	8
SDRAM Auto Refresh Timer bits 7-0							
7	6	5	4	3	2	1	0

bits 15-0 SDRAM Auto Refresh Timer bits [15:0]
These bits control the period between SDRAM auto refresh cycles. These bits must not be set to a value less than 4h.

period between auto refresh cycles = REG[1C0Ch] bits 15-0 + 9 MEMCLK periods

The number of REF cycles per MEMCKE high pulse is controlled by REG[1C04h] bits 11-8.

REG[1C0Eh] is Reserved

This register is Reserved and should not be written.

REG[1C10h] SDRAM Mode Setting Value Register Default = 0000h							Read Only
SDRAM Mode Setting Value bits 15-8							
15	14	13	12	11	10	9	8
SDRAM Mode Setting Value bits 7- 0							
7	6	5	4	3	2	1	0

bits 15-0 SDRAM Mode Setting Value bits [15:0] (Read Only)
These bits indicate the value that is written to the SDRAM mode register. The mode register specifies the operational parameters for the external SDRAM.

<table>
<tr><td colspan="8"><b>REG[1C12h] Mobile SDRAM Configuration Register</b>
Default = 4000h Read/Write</td></tr>
<tr><td colspan="2">TRSC bits 1-0</td><td colspan="6">n/a</td></tr>
<tr><td>15</td><td>14</td><td>13</td><td>12</td><td>11</td><td>10</td><td>9</td><td>8</td></tr>
<tr><td>Mobile Select</td><td>n/a</td><td>Auto Temperature Compensated Self Refresh Enable</td><td colspan="2">Reserved</td><td colspan="3">Partial Array Self Refresh Select bits 2-0</td></tr>
<tr><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
</table>

bits 15-14 TRSC bits [1:0]
These bits specify the number of clocks between the Mode Register Set Cycles (clocks between Mode Setting and Extended Mode Setting) when mobile SDRAM is selected, REG[1C12h] bit 7 = 1b.

Table 10-70 : Mode Register Set Cycle

REG[1C12h] bits 15-14	TRSC
00b	Reserved
01b	2 Clocks
10b	3 Clocks
11b	4 Clocks

bit 7 Mobile Select
This bit selects whether normal SDRAM or mobile SDRAM is used. When this bit is set, the Extended Mode Setting register is programmed for mobile SDRAM after the SDRAM is initialized (see REG[1C02h] bit 0).
When this bit = 0b, normal SDRAM is selected.
When this bit = 1b, mobile SDRAM is selected.

bit 5 Auto Temperature Compensated Self Refresh Enable
This bit controls the auto temperature compensated self refresh function.
When this bit = 0b, auto temperature compensated self refresh is disabled.
When this bit = 1b, auto temperature compensated self refresh is enabled.

bits 4-3 Reserved
These bits must be set to 00b.

bits 2-0 Partial Array Self Refresh Select bits [2:0]
These bits control which banks of the mobile SDRAM are self refreshed. The value of these bits should be programmed according to the requirements of the mobile SDRAM used. For details, refer to the Mobile SDRAM Specification.

Table 10-71 : Partial Array Self Refresh Selection

REG[1C12h] bits 2-0	Partial Array Self Refresh
000b	All Banks
001b	Bank A and Bank B (BA1 = 0)
010b	Bank A (BA1=0 and BA0 = 0)
011b - 111b	Reserved

REG[1C14h] Mobile SDRAM Extended Mode Setting Register
Default = 0000h Read Only

Mobile SDRAM Extended Mode Setting bits 15-8							
15	14	13	12	11	10	9	8
Mobile SDRAM Extended Mode Setting bits 7-0							
7	6	5	4	3	2	1	0

bits 15-0 Mobile SDRAM Extended Mode Setting bits [15:0] (Read Only)
These bits indicates the value that is written to the SDRAM extended mode register. The extended mode register specifies the operational parameters for the external mobile SDRAM.

10.4.11 Camera Interface Registers

Note

The pins used by the camera interface are multiplexed with GPIO function pins. Therefore, before enabling the camera interface, the appropriate GPIO pins must be configured for use by the camera interface. For a summary of GPIO pin usage, see Section 5.6, “GPIO Pin Mapping” on page 50.

REG[2000h] Camera1 Clock Setting Register
Default = 0000h Read/Write

n/a							
15	14	13	12	11	10	9	8
n/a			Camera1 Clock Divide Select bits 4-0				
7	6	5	4	3	2	1	0

bits 4-0 Camera1 Clock Divide Select bits [4:0]
These bits specify the divide ratio used to generate the Camera1 Clock Output (CM1CLKOUT) from the System Clock.

Table 10-72: Camera1 Clock Divide Ratio Selection

REG[2000h] bits 4-0	Camera1 Clock Divide Ratio	REG[2000h] bits 4-0	Camera1 Clock Divide Ratio
00000b (default)	Reserved	10000b	17:1
00001b	Reserved	10001b	18:1
00010b	3:1	10010b	19:1
00011b	4:1	10011b	20:1
00100b	5:1	10100b	21:1
00101b	6:1	10101b	22:1
00110b	7:1	10110b	23:1
00111b	8:1	10111b	24:1
01000b	9:1	11000b	25:1
01001b	10:1	11001b	26:1
01010b	11:1	11010b	27:1
01011b	12:1	11011b	28:1
01100b	13:1	11100b	29:1
01101b	14:1	11101b	30:1
01110b	15:1	11110b	31:1
01111b	16:1	11111b	32:1

REG[2002h] Camera1 Signal Setting Register
Default = 0000h Read/Write

n/a							
15	14	13	12	11	10	9	8
n/a	Camera1 Interface Select	Camera1 Clock Mode Select	Camera1 YUV Data Format Select bits 1-0		CM1HREF Active Select	CM1VREF Active Select	CM1CLKIN Active Select
7	6	5	4	3	2	1	0

bit 6 Camera1 Interface Select
This bit selects the Camera1 interface type.
When this bit = 0b, the Camera1 interface is configured for YUV 4:2:2 8-bit.
When this bit = 1b, the Camera1 interface is configured for YUV 4:2:2 16-bit.

bit 5 Camera1 Clock Mode Select
This bit determines the source of the clock used to sample incoming YUV data on CM1DAT[7:0] for the Camera1 interface.
When this bit = 0b, the external input clock (CM1CLKIN) from the camera interface is used to sample incoming YUV data. (default)
When this bit = 1b, the internal camera clock output (CM1CLKOUT) is used to sample incoming YUV data. When in this camera clock mode, the Camera1 Clock Divide Select bits (REG[2000h] bits 4-0) must be set to a ratio greater than 2:1.

bits 4-3 Camera1 YUV Data Format Select bits [1:0]
These bits specify the YUV data sequence order for the Camera1 interface.

Table 10-73: YUV Data Format Selection

REG[2002h] bits 4-3	YUV Data Format (8-bit format)	YUV Data Format (16-bit format)
00b (default)	(1st) UYVY (last)	(1st cam1) U V (last) (1st cam2) Y Y (last)
01b	(1st) VYUY (last)	(1st cam1) V U (last) (1st cam2) Y Y (last)
10b	(1st) YUYV (last)	(1st cam1) Y Y (last) (1st cam2) U V (last)
11b	(1st) YVYU (last)	(1st cam1) Y Y (last) (1st cam2) V U (last)

bit 2 CM1HREF Active Select
This bit selects the polarity of CM1HREF for the Camera1 interface.
When this bit = 0b, the Camera1 hsync (CM1HREF) is active low and CM1HREF high means data is valid.
When this bit = 1b, the Camera1 hsync (CM1HREF) is active high and CM1HREF low means data is valid.

Note

This bit can only be modified while the camera interface is disabled (REG[2010h] bit 0 = 0b).

bit 1 CM1VREF Active Select
This bit selects the polarity of CM1VREF for the Camera1 interface.
When this bit = 0b, the Camera1 vsync (CM1VREF) is active low and CM1VREF high means data is valid.
When this bit = 1b, the Camera1 vsync (CM1VREF) is active high and CM1VREF low means data is valid.

Note

This bit can only be modified while the camera interface is disabled (REG[2010h] bit 0 = 0b).

bit 0 CM1CLKIN Active Select
This bit selects which edge of CM1CLKIN is used to latch input data. CM1CLKIN is used for the Camera1 interface.
When this bit = 0b, the S1D13513 latches input data at the rising edge of CM1CLKIN.
When this bit = 1b, the S1D13513 latches input data at the falling edge of CM1CLKIN.

Note

This bit can only be modified while the camera interface is disabled (REG[2010h] bit 0 = 0b).

REG[2004h] Camera2 Clock Setting Register

Default = 0000h Read/Write

n/a							
15	14	13	12	11	10	9	8
n/a			Camera2 Clock Divide Select bits 4-0				
7	6	5	4	3	2	1	0

bits 4-0 Camera2 Clock Divide Select bits [4:0]
These bits specify the divide ratio used to generate the Camera2 Clock Output (CM2CLKOUT) from the System Clock.

Table 10-74: Camera2 Clock Divide Ratio Selection

REG[2004h] bits 4-0	Camera2 Clock Divide Ratio
00000b	Reserved
00001b	Reserved
00010b	3:1
00011b	4:1
00100b	5:1
00101b	6:1
00110b	7:1
00111b	8:1
01000b	9:1
01001b	10:1
01010b	11:1
01011b	12:1
01100b	13:1
01101b	14:1
01110b	15:1
01111b	16:1
10000b	17:1
10001b	18:1
10010b	19:1
10011b	20:1
10100b	21:1
10101b	22:1
10110b	23:1
10111b	24:1
11000b	25:1
11001b	26:1
11010b	27:1
11011b	28:1
11100b	29:1
11101b	30:1
11110b	31:1
11111b	32:1

REG[2006h] Camera2 Signal Setting Register Default = 0000h							Read/Write
n/a							
15	14	13	12	11	10	9	8
Reserved		Camera2 Clock Mode Select	Camera2 YUV Data Format Select bits 1-0		CM2HREF Active Select	CM2VREF Active Select	CM2CLKIN Active Select
7	6	5	4	3	2	1	0

bits 7-6 Reserved
The default value for these bits is 00b.

bit 5 Camera2 Clock Mode Select
This bit determines the source of the clock used to sample incoming YUV data on CM2DAT[7:0] for the Camera2 interface.
When this bit = 0b, the external input clock (CM2CLKIN) from the camera interface is used to sample incoming YUV data (default).
When this bit = 1b, the internally divided system clock (CM2CLKOUT) is used to sample incoming YUV data.

bits 4-3 Camera2 YUV Data Format Select bits [1:0]
These bits specify the YUV data sequence order for the Camera2 interface.

Table 10-75: YUV Data Format Selection

REG[2006h] bits 4-3	YUV Format
00b	(1st) UYVY (last)
01b	(1st) VYUY (last)
10b	(1st) YUYV (last)
11b	(1st) YVYU (last)

bit 2 CM2HREF Active Select
This bit selects the polarity of CM2HREF for the Camera2 interface.
When this bit = 0b, the Camera2 hsync (CM2HREF) is active low and CM2HREF high means data is valid.
When this bit = 1b, the Camera2 hsync (CM2HREF) is active high and CM2HREF low means data is valid.

Note
This bit can only be modified while the camera interface is disabled (REG[2010h] bit 0 = 0b).

bit 1 CM2VREF Active Select
This bit selects the polarity of CM2VREF for the Camera2 interface.
When this bit = 0b, the Camera2 vsync (CM2VREF) is active low and CM2VREF high means data is valid.
When this bit = 1b, the Camera2 vsync (CM2VREF) is active high and CM2VREF low means data is valid.

Note
This bit can only be modified while the camera interface is disabled (REG[2010h] bit 0 = 0b).

bit 0 CM2CLKIN Active Select
This bit selects which edge of CM2CLKIN is used to latch input data. CM2CLKIN is used for the Camera2 interface.
When this bit = 0b, the S1D13513 latches input data at the rising edge of CM2CLKIN.
When this bit = 1b, the S1D13513 latches input data at the falling edge of CM2CLKIN.

Note
This bit can only be modified while the camera interface is disabled (REG[2010h] bit 0 = 0b).

REG[2008h] through REG[200Eh] are Reserved

These registers are Reserved and should not be written.

REG[2010h] Camera Mode Setting Register
Default = 0000h Read/Write

Reserved	Reserved	Reserved	Reserved	Reserved		Reserved	YUV Data Type
15	14	13	12	11	10	9	8
ITU-R BT656 Enable	Camera Mode Select bits 2-0			Clock Output Port Select bits 2-0			Camera Interface Enable
7	6	5	4	3	2	1	0

bit 15 Reserved
The default value for this bit is 0b.

bit 14 Reserved
The default value for this bit is 0b.

bit 13 Reserved
The default value for this bit is 0b.

bit 12 Reserved
The default value for this bit is 0b.

bits 11-10 Reserved
These bits must be set to 00b.

bit 9 Reserved
The default value for this bit is 0b.

bit 8 YUV Data Type
This bit specifies whether an UV offset is applied to the incoming camera data and must be configured based on the YUV data type of the camera. Typically, camera modules output with UV offset, so this bit should be set to 0b.

Table 10-76: Camera Input YUV Data Type Selection

REG[2010h] bit 8	YUV Data Type	Data Range 1	Data Range 2
0b (default)	Straight Binary	$0 \le U \le 255$ $0 \le V \le 255$	$16 \le Cb \le 240$ $16 \le Cr \le 240$
1b	Offset Binary	$-128 \le U \le 127$ $-128 \le V \le 127$	$-112 \le Cb \le 112$ $-112 \le Cr \le 112$

bit 7 ITU-R BT656 Enable
This bit controls the active camera interface type.
When this bit = 0b, the normal camera interface is active. In this mode the hsync, vsync, clock, and data signals are independent. (default)
When this bit = 1b, the ITU-R BT656 camera interface is active. In this mode the hsync and vsync signal information is embedded in the data signals.

bits 6-4 Camera Mode Select bits [2:0]
These bits select the active camera mode.

Table 10-77: Camera Mode Selection

REG[2010h] bits 6-4	Active Camera Mode
000b	Camera1 Interface Input is Active
001b	Camera2 Interface Input is Active
010b - 111b	Reserved

bits 3-1 Clock Output Port Select bits [2:0]
These bits select the active clock output port.

Table 10-78: Clock Output Port Selection

REG[2010h] bits 3-1	Active Clock Output Port
000b	Same Active Port as selected by REG[2010h] bits 6-4
001b	Camera1 Output Port Active Only
010b	Camera2 Output Port Active Only
011b	Both Camera1 and Camera2 Output Port Active
100b	Clock Output Inactive
101b - 111b	Reserved

bit 0 Camera Interface Enable
This bit controls the camera interface. When the camera interface is disabled, the clock is stopped and the camera clock output is disabled.
When this bit = 0b, the camera interface is disabled (low output). (default)
When this bit = 1b, the camera interface is enabled.

Note

1. Setting this bit to 0b when CMCLKOUT is toggling causes the CMCLKOUT signal to be driven low immediately.
2. When disabling the camera interface using this bit, partial camera data may not be cleared, and may corrupt future camera/resizer operation. When disabling the camera, it is recommended to use the Camera Interface Software Reset bit (REG[2014h] bit 0 = 1b), which both initializes and disables the camera interface. This method clears any partial data associated with the camera interface.

<table>
<tr><th colspan="8">REG[2012h] Camera Frame Setting Register
Default = 0000h Read/Write</th></tr>
<tr><td colspan="3">n/a</td><td>Reserved</td><td colspan="3">n/a</td><td>Raw JPEG Capture Mode Enable</td></tr>
<tr><td>15</td><td>14</td><td>13</td><td>12</td><td>11</td><td>10</td><td>9</td><td>8</td></tr>
<tr><td>Frame Capture Interrupt Select</td><td>Single Frame Capture Select</td><td>Reserved</td><td colspan="3">Frame Sampling Mode bits 2-0</td><td>Frame Capture Interrupt Polarity</td><td>Frame Capture Interrupt Enable</td></tr>
<tr><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
</table>

bit 12 Reserved
The default value for this bit is 0b.

bit 8 Raw JPEG Capture Mode Enable
This bit controls Raw JPEG Capture Mode. Some camera modules can output JPEG encoded data directly from the camera. To allow the S1D13513 to capture this JPEG data, this bit must be set to 1b.
When this bit = 0b, raw JPEG capture mode is disabled.
When this bit = 1b, raw JPEG capture mode is enabled.

Note

1. This bit reflect while VBLANK and data capture are stopped. Raw JPEG capture mode cannot be enabled while the camera interface is disabled, REG[2010h] bit 0 = 0b.
2. The strobe function (REG[2020h] ~ REG[2024h]) cannot be used while raw JPEG capture mode is enabled.
3. When raw JPEG data is captured, it is padded to ensure that the file size is a multiple of 32 bits. The padding is applied as follows.

Table 10-79: Raw JPEG Capture Data Padding

File Size	Padding File Format
4n bytes	(1st) Input Data + FFFF_FFFFh (last)
4n + 1 bytes	(1st) Input Data + FF_FFFFh (last)
4n + 2 bytes	(1st) Input Data + FFFFh (last)
4n + 3 bytes	(1st) Input Data + FFh (last)

bit 7 Frame Capture Interrupt Select
This bit selects when the Frame Capture Interrupt is asserted (see REG[2012h] bit 0).
When this bit = 0b, the Frame Capture Interrupt is asserted while a valid frame is being captured.
When this bit = 1b, the Frame Capture Interrupt is asserted after a valid frame has been captured and the camera interface has stopped capturing data.

bit 6 Single Frame Capture Select
This bit controls the frame capture mode for single frame captures. This bit must not be changed while the camera interface is enabled, REG[2010h] bit 0 = 1b.
When this bit = 0b, the image from the camera interface is continuously captured.
When this bit = 1b, the next frame is captured and then the camera interface is disabled.

bit 5 Reserved
The default value for this bit is 0b.

bits 4-2 Frame Sampling Mode bits [2:0]
These bits control the camera data sampling rate, in frames.

Table 10-80: Frame Sampling Control Selection

REG[2012h] bits 4-2	Frame Sampling Mode
000b	All frames are sampled
001b	1 of every 2 frames is sampled
010b	1 of every 3 frames is sampled
011b	1 of every 4 frames is sampled
100b	1 of every 5 frames is sampled
101b	1 of every 6 frames is sampled
110b	1 of every 7 frames is sampled
111b	No frames are sampled

bit 1 Frame Capture Interrupt Polarity
This bit controls the assertion timing of the Frame Capture Interrupt.
When this bit = 0b, the Frame Capture Interrupt is asserted when CMVREF is active. (default)
When this bit = 1b, the Frame Capture Interrupt is asserted when CMVREF is inactive.

bit 0 Frame Capture Interrupt Enable
This bit controls whether a Frame Capture Interrupt is generated.
When this bit = 0b, the frame capture interrupt is disabled. (default)
When this bit = 1b, the frame capture interrupt is enabled.

REG[2014h] Camera Control Register
Default = not applicable Write Only

n/a					Raw JPEG Capture Mode Error Flag Clear	ITU-R BT656 Error Flag 1 Clear	ITU-R BT656 Error Flag 0 Clear
15	14	13	12	11	10	9	8
n/a				Frame Capture Stop	Frame Capture Start	Frame Capture Interrupt Flag Clear	Camera Interface Software Reset
7	6	5	4	3	2	1	0

bit 10 Raw JPEG Capture Mode Error Flag Clear (Write Only)
This bit only has an effect when Raw JPEG Capture Mode is enabled, REG[2012h] bit 8 = 1b.
Writing a 0b to this bit has no hardware effect.
Writing a 1b to this bit clears the Raw JPEG Capture Mode Error Flag, REG[2016h] bit 10.

bit 9 ITU-R BT656 Error Flag 1 Clear (Write Only)
This bit only has an effect when ITU-R BT656 interface mode is active, REG[2010h] bit 7 = 1b.
Writing a 0b to this bit has no hardware effect.
Writing a 1b to this bit clears the ITU-R BT656 Error Flag 1, REG[2016h] bit 9.

bit 8 ITU-R BT656 Error Flag 0 Clear (Write Only)
This bit only has an effect when ITU-R BT656 interface mode is active, (REG[2010h] bit 7 = 1b.
Writing a 0b to this bit has no hardware effect.
Writing a 1b to this bit clears the ITU-R BT656 Error Flag 0, REG[2016h] bit 8.

bit 3 Frame Capture Stop (Write Only)
This bit stops image frame capturing from the camera interface once the current camera frame has ended. The current camera capture state is indicated by the Frame Capture Start/Stop Flag, REG[2016h] bit 2.
Writing a 0b to this bit has no hardware effect.
Writing a 1b to this bit stops image frame capturing after the current camera frame.

Note

When the camera interface is actively capturing frame data, a Frame Capture Stop must be issued in order to allow the current camera frame to finish before disabling the camera interface.

bit 2 Frame Capture Start (Write Only)
This bit starts image frame capturing from the camera interface on the next camera frame. The current camera capture state is indicated by the Frame Capture Start/Stop Flag, REG[2016h] bit 2.
Writing a 0b to this bit has no hardware effect.
Writing a 1b to this bit starts image frame capturing on the next camera frame.

Note

1. The default frame capture state is “start” after a hardware reset, or after a camera interface software reset (REG[2014h] bit 0) when the Single Frame Capture Select bit is set for continuous capture (REG[2012h] bit 6 = 0b).
2. For YUV capture modes (REG[2800h] bits 3-1 = 011b and 111b), this bit must be written twice to capture a frame when single capture mode is selected, REG[2012h] bit 6 = 1b.

bit 1 Frame Capture Interrupt Flag Clear (Write Only)
This bit clears the Frame Capture Interrupt Status bit, REG[2016h] bit 1.
Writing a 0b to this bit has no hardware effect.
Writing a 1b to this bit clears the Frame Capture Interrupt Status.

bit 0 Camera Interface Software Reset (Write Only)
This bit initializes the camera interface logic and disables the camera interface (REG[2010h] bit 0). Camera interface registers other than the Camera Status Register (REG[2016h]) are not affected by this bit.
Writing a 0b to this bit has no hardware effect.
Writing a 1b to this bit initializes and disables the camera interface.

Note

It is recommended that the camera interface be disabled using this bit.

REG[2016h] Camera Status Register Default = 0004h							Read Only
n/a					Raw JPEG Capture Mode Error Flag	ITU-R BT656 Error Flag 1	ITU-R BT656 Error Flag 0
15	14	13	12	11	10	9	8
n/a	CMVREF Status	Effective Strobe Frame Status	Effective Frame Status	Frame Capture Busy Status	Frame Capture Start/Stop Flag	Frame Capture Interrupt Flag	n/a
7	6	5	4	3	2	1	0

bit 10

Raw JPEG Capture Mode Error Flag (Read Only)
This bit only has an effect when Raw JPEG Capture Mode is enabled, REG[2012h] bit 8 = 1b.
When this bit = 0b, no error has occurred.
When this bit = 1b, an error with the JPEG data is indicated by the camera (HSYNC high during active VSYNC)

To clear this flag, write a 1b to the Raw JPEG Capture Mode Error Flag Clear bit (REG[2014h] bit 10 = 1b).

bit 9

ITU-R BT656 Error Flag 1 (Read Only)
This bit only has an effect when ITU-R BT656 interface mode is active, REG[2010h] bit 7 = 1b.
When this bit = 0b, no error has occurred.
When this bit = 1b, a 2-bit error is detected on the reference decode operation.

To clear this flag, write a 1b to the ITU-R BT565 Error Flag 1 Clear bit (REG[2014h] bit 9 = 1b).

bit 8

ITU-R BT656 Error Flag 0 (Read Only)
This bit only has an effect when ITU-R BT656 interface mode is active, REG[2010h] bit 7 = 1b.
When this bit = 0b, no error correction has occurred.
When this bit = 1b, a 1-bit error correction is detected on the reference decode operation.

To clear this flag, write a 1b to the ITU-R BT565 Error Flag 0 Clear bit (REG[2014h] bit 8 = 1b).

bit 6

CMVREF Status (Read Only)
This bit indicates the current condition of CMVREF input from the camera interface.
When this bit = 0b, CMVREF input is low.
When this bit = 1b, CMVREF input is high.

bit 5

Effective Strobe Frame Status (Read Only)
This bit indicates whether the current frame is a “valid data capture” frame, after a strobe control output is triggered by a frame capture stop command (REG[2014h] bit 3 = 1b). This bit only has an effect while the camera interface is running in repeat capture mode (REG[2012h] bit 6 = 0b). The valid data capture frame is determined by the Strobe Capture Delay bits (REG[2024h] bits 7-4). For further information, see Section 22.3, “Strobe Control Signal” on page 469.
When this bit = 0b, there is no valid data.
When this bit = 1b, valid data is currently forwarding.

bit 4 Effective Frame Status (Read Only)
This bit indicates whether the current frame from the camera interface is an "effective" frame (captured by the camera interface) based on the Frame Sampling Mode bits (REG[2012h] bit 4-2).
When this bit = 0b, an effective frame is not occurring.
When this bit = 1b, an effective frame is occurring.

The following diagram shows an example of the Effective Frame Status bit where the Frame Sampling Mode bits are set for 1 frame sampled for every 3 frames (REG[2012h] bits 4-2 = 010b).

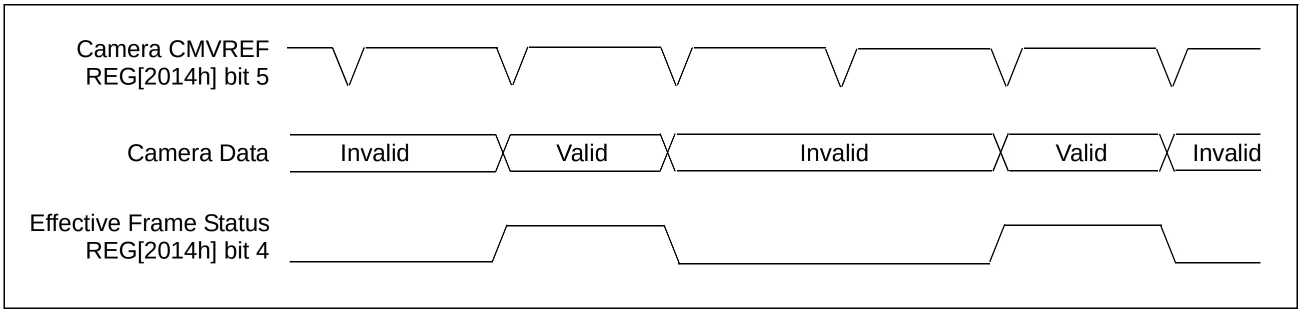


Figure 10-3: Effective Frame Status Bit Example

bit 3 Frame Capture Busy Status (Read Only)
This bit indicates the status of frame capturing from the camera interface.
When this bit = 0b, frames are not being captured.
When this bit = 1b, frames are being captured.

bit 2 Frame Capture Start/Stop Status (Read Only)
This bit monitors the camera frame start/stop bits (REG[2014h] bits 3-2).
When this bit = 0b, the frame capturing start command has not been asserted.
When this bit = 1b, the frame capturing start command has been asserted.

Note

The default frame capture state, after a hardware reset or after a camera interface software reset (REG[2014h] bit 0), is "start". If a frame capture state of "stop" is required, the Frame Capture Stop bit must be set (REG[2014h] bit 4 = 1b) before enabling the camera interface (REG[2010h] bit 0).

bit 1 Frame Capture Interrupt Status (Read Only)
This bit indicates the state of the Frame Capture Interrupt.
When this bit = 0b, a frame capture interrupt has not occurred.
When this bit = 1b, a frame capture interrupt has occurred.

REG[2020h] Strobe Control Signal Output Delay Setting Register Default = 0000h							Read/Write
Strobe Line Delay Timing bits 15-8							
15	14	13	12	11	10	9	8
Strobe Line Delay Timing bits 7-0							
7	6	5	4	3	2	1	0

bits 15-0 Strobe Line Delay Timing bits [15:0]
When the strobe is enabled (REG[2024h] bit 0 = 1b), these bits specify the delay from the first falling edge of CMHREF after CMVREF active to the beginning of the strobe control signal (CMSTROUT) output, in number of lines from the camera interface (CMHREF). For details on the Strobe Control Signal, see Section 22.3, “Strobe Control Signal” on page 469.

REG[2022h] Strobe Control Signal Pulse Width Setting Register Default = 0000h							Read/Write
Strobe Pulse Width bits 15-8							
15	14	13	12	11	10	9	8
Strobe Pulse Width bits 7-0							
7	6	5	4	3	2	1	0

bits 15-0 Strobe Pulse Width bits [15:0]
When the strobe is enabled (REG[2024h] bit 0 = 1b), these bits specify the strobe control signal (CMSTROUT) pulse width, in number of lines from the camera interface (CMHREF). For details on the Strobe Control Signal, see Section 22.3, “Strobe Control Signal” on page 469.

Strobe Pulse Width = REG[2022h] bits 15-0 + 1 lines

REG[2024h] Strobe Setting Register Default = 0000h							Read/Write
Camera Strobe Request Flag (RO)	n/a			Camera Strobe Enable	n/a		
15	14	13	12	11	10	9	8
Strobe Capture Delay bits 3-0				Reserved	Reserved	Strobe Active Select	Strobe Port Enable
7	6	5	4	3	2	1	0

bit 15 Camera Strobe Request Flag (Read Only)
This bit indicates whether the camera has requested a strobe. The camera can request a strobe by holding HYSNC high and VSYNC low for 4 camera clock periods. This bit is cleared at the end of each camera frame. This bit can be used to detect when the camera is requesting a flash to be fired.
When this bit = 0b, a camera strobe has not been requested by the camera.
When this bit = 1b, a camera strobe has been requested by the camera.

bit 11 Camera Strobe Enable
This bit controls how the strobe function is triggered. When this bit is enabled, the camera can trigger the strobe by holding HSYNC high and VSYNC low for 4 camera clock periods. This allows the camera to control when an external flash is fired.
When this bit = 0b, camera control of the strobe function is disabled.
When this bit = 1b, camera control of the strobe function is enabled.

bits 7-4 Strobe Capture Delay bits [3:0]
These bits control the timing of the camera frame captured using the strobe control signal output. When the strobe is enabled (REG[2024h] bit 0 = 1b) and the camera interface is in repeat capture mode (REG[2012h] bit 6 = 0b), these bits specify the number of frames delayed from the strobe control signal output to:

- the valid camera frame capture (for YUV Capture)
- the last frame captured by the camera interface (for view mode)

These bits have no effect and the delay is always configured for “no delay” when the camera interface is in single frame capture mode (REG[2012h] bit 6 = 1b).

The following table shows the possible delay values, in frames.

Table 10-81: Strobe Capture Delay Control

REG[2024h] bits 7-4	Delay Value	REG[2024h] bits 7-4	Delay Value
0000b (default)	No Delay	1000b	8 Frames
0001b	1 Frame	1001b	9 Frames
0010b	2 Frames	1010b	10 Frames
0011b	3 Frames	1011b	11 Frames
0100b	4 Frames	1100b	12 Frames
0101b	5 Frames	1101b	13 Frames
0110b	6 Frames	1110b	14 Frames
0111b	7 Frames	1111b	15 Frames

bit 3 Reserved
The default value for this bit is 0b.

bit 2 Reserved
The default value for this bit is 0b.

bit 1 Strobe Active Select
This bit determines the active polarity of the strobe control signal (CMSTROUT) and only has an effect when the output mode of the Strobe Port Select bit is configured for the strobe, REG[2024h] bit 0 = 1b.
When this bit = 0b, the strobe control signal is active low. (default)
When this bit = 1b, the strobe control signal is active high.

bit 0 Strobe Port Enable
This bit enables the strobe output (CMSTROUT) when GPIOB7 is configured for CMSTROUT (Non-GPIO function #2), REG[0C06h] bits 15-14 = 11b.
When this bit = 0b, strobe output is disabled.
When this bit = 1b, strobe output (flash) is enabled. For further information on this function, see Section 22.4 “Strobe Control Signal” on page 450. When strobe output is enabled, CMSTROUT outputs a strobe pulse triggered by the following:

- the Frame Capture Stop bit for repeat capture mode (REG[2014h] bit 2 = 1b)
- the Frame Capture Start bit for single frame capture mode (REG[2014h] bit 3 =1b)

Note
To use the CMSTROUT function, GPIOB7 must be configured for “Non-GPIO Function #2”, REG[0C06h] bits 15-14 = 11b.

REG[2028h] through REG[202Eh] are Reserved

These registers are Reserved and should not be written.

10.4.12 Resizer Operation Registers

Note

The Resizer Operation registers must not be changed while receiving data from the camera interface or host interface.

REG[2430h] Global Resizer Control Register
Default = 0000h Read/Write

n/a					Resizer Frame Reduction	Reserved	Reserved
15	14	13	12	11	10	9	8
n/a		JPEG Camera Data Input Enable	Reserved	Reserved	n/a	Camera Display Control bits 1-0	
7	6	5	4	3	2	1	0

bit 10 Resizer Frame Reduction
This bit controls frame reduction in the resizer block.
When this bit = 0b, the resizer performs no reduction.
When this bit = 1b, the resizer performs frame reduction by using only every second frame.

bit 9 Reserved
The default value for this bit is 0b.

bit 8 Reserved
The default value for this bit is 0b.

bit 5 JPEG Camera Data Input Enable
This bit controls whether the resizer can receive JPEG encoded data from a JPEG capable camera.
When this bit = 0b, JPEG encoded data cannot be input.
When this bit = 1b, JPEG encoded data can be input.

bit 4 Reserved
The default value for this bit is 0b.

bit 3 Reserved
The default value for this bit is 0b.

bits 1-0 Camera Display Control bits [1:0]
These bits control how camera data is displayed when YUV Capture mode is enabled (REG[2800h] bits 3-1 = 011b or 111b)..

Table 10-82 : Camera Display Control Selection

REG[2430h] bits 1-0	Function
00b	**YUV Capture:** YUV data from the camera interface is continuously written to the YUV FIFO and converted YUV data (YUV to RGB Converter) is continuously written to the display buffer.
01b	**YUV Capture:** YUV data from the camera interface is continuously written to the YUV FIFO. When the shutter is enabled, REG[280Ah] bit 0 = 1b, camera data is written to the display buffer. When the shutter is disabled, REG[280Ah] bit 0 = 0b, camera data is not written to the display buffer.
10b	**Reserved**
11b	Reserved.

REG[2432h] through REG[243Eh] are Reserved

These registers are Reserved and should not be written.

View (Display) Resizer Registers

.

REG[2440h] View Resizer Control Register
Default = 0000h Read/Write

n/a							
15	14	13	12	11	10	9	8
View Resizer Software Reset (WO)	n/a				View Resizer Independent Horizontal/Vertical Scaling Enable	View Resizer Register Update VSYNC Enable	View Resizer Enable
7	6	5	4	3	2	1	0

bit 7 View Resizer Software Reset (Write Only)
This bit performs a software reset of the view resizer logic.
Writing a 0b to this bit has no hardware effect.
Writing a 1b to this bit while the resizers are activated (REG[2440h] bit 0 = 1b or REG[2460h] bit 0 = 1b) performs a software reset of the view resizer.

bit 2 View Resizer Independent Horizontal/Vertical Scaling Enable
This bit determines whether the view resizer uses the same scaling rate for the horizontal and vertical directions, or uses independent scaling rates.
When this bit = 0b, the horizontal and vertical scaling rates are the same. Both horizontal and vertical scaling rates are controlled by REG[244Ch] bits 7-0.
When this bit = 1b, the horizontal and vertical scaling rates can be selected independently. Horizontal scaling rate is controlled by REG[244Ch] bits 7-0 and vertical scaling rate is controlled by REG[244Ch] bits 15-8.

bit 1 View Resizer Register Update VSYNC Enable
This bit determines whether the view resizer logic uses new values programmed to the view resizer registers immediately, or only once the next VSYNC occurs.
When this bit = 0b, the view resizer uses the new register values immediately.
When this bit = 1b, the view resizer uses the previous register values until the next VSYNC occurs when it uses the new register values.

Note

When REG[2800h] bits 3-1 = 001b, this bit must be set to 0b.

bit 0 View Resizer Enable
This bit controls the view resizer.
When this bit = 0b, the view resizer is disabled.
When this bit = 1b, the view resizer is enabled.

Note

1. The View Resizer and the Capture Resizer (REG[2460h] bit 0) must not be enabled at the same time.
2. When this bit and the Capture Resizer Enable bit (REG[2460h] bit 0) are both set to 0b, the clock to the resizer block is automatically stopped.

REG[2444h] View Resizer Start X Position Register
Default = 0000h Read/Write

n/a					View Resizer Start X Position bits 10-8		
15	14	13	12	11	10	9	8
View Resizer Start X Position bits 7-0							
7	6	5	4	3	2	1	0

bits 10-0 View Resizer Start X Position bits [10:0]
These bits determine the X start position for the View Resizer. These bits must be programmed according to the restrictions in Section 14, “Resizers” on page 393.

Note

The resizer output size, after trimming and scaling, must be an even number of pixels.

REG[2446h] View Resizer Start Y Position Register
Default = 0000h Read/Write

n/a					View Resizer Start Y Position bits 10-8		
15	14	13	12	11	10	9	8
View Resizer Start Y Position bits 7-0							
7	6	5	4	3	2	1	0

bits 10-0 View Resizer Start Y Position bits [10:0]
These bits determine the Y start position for the View Resizer. These bits must be programmed according to the restrictions in Section 14, “Resizers” on page 393.

Note

The resizer output size, after trimming and scaling, must be an even number of pixels.

REG[2448h] View Resizer End X Position Register
Default = 027Fh Read/Write

n/a					View Resizer End X Position bits 10-8		
15	14	13	12	11	10	9	8
View Resizer End X Position bits 7-0							
7	6	5	4	3	2	1	0

bits 10-0 View Resizer End X Position bits [10:0]
These bits determine the X end position for the View Resizer. These bits must be programmed according to the restrictions in Section 14, “Resizers” on page 393.

Note
The resizer output size, after trimming and scaling, must be an even number of pixels.

REG[244Ah] View Resizer End Y Position Register
Default = 01DFh Read/Write

n/a					View Resizer End Y Position bits 10-8		
15	14	13	12	11	10	9	8
View Resizer End Y Position bits 7-0							
7	6	5	4	3	2	1	0

bits 10-0 View Resizer End Y Position bits [10:0]
These bits determine the Y end position for the View Resizer. These bits must be programmed according to the restrictions in Section 14, “Resizers” on page 393.

Note
The resizer output size, after trimming and scaling, must be an even number of pixels.

REG[244Ch] View Resizer Scaling Rate Register
Default = 8080h Read/Write

View Resizer Vertical Scaling Rate bits 7-0							
15	14	13	12	11	10	9	8
View Resizer Horizontal Scaling Rate bits 7-0							
7	6	5	4	3	2	1	0

bits 15-8 View Resizer Vertical Scaling Rate bits [7:0]
These bits determine the view resizer vertical scaling rate when REG[2440h] bit 2 = 1b. Not all scaling rates are available for all scaling modes (see REG[244Eh]).

Table 10-83: View Resizer Vertical Scaling Rate Selection

REG[244Ch] bits 15-8	**View Resizer Vertical Scaling Rate**			
	REG[244Eh] bits 1-0 = 00b	**REG[244Eh] bits 1-0 = 01b**	**REG[244Eh] bits 1-0 = 10b**	**REG[244Eh] bits 1-0 = 11b**
0000 0000b	Reserved	Reserved	Reserved	Reserved
0000 0001b	n/a	1/128	1/128	Reserved
0000 0010b	n/a	2/128	2/128	Reserved
0000 0011b	n/a	3/128	3/128	Reserved
0000 0100b	n/a	4/128	4/128	Reserved
0000 0101b	n/a	5/128	5/128	Reserved
0000 0110b	n/a	6/128	6/128	Reserved
0000 0111b	n/a	7/128	7/128	Reserved
0000 1000b	n/a	8/128	8/128	Reserved
0000 1001b	n/a	9/128	9/128	Reserved
0000 1010b	n/a	10/128	10/128	Reserved
0000 1011b	n/a	11/128	11/128	Reserved
0000 1100b	n/a	12/128	12/128	Reserved
0000 1101b	n/a	13/128	13/128	Reserved
0000 1110b	n/a	14/128	14/128	Reserved
0000 1111b	n/a	15/128	15/128	Reserved
0001 0000b	n/a	16/128	16/128	Reserved
0001 0001b	n/a	17/128	17/128	Reserved
0001 0010b	n/a	18/128	18/128	Reserved
0001 0011b	n/a	19/128	19/128	Reserved
0001 0100b	n/a	20/128	20/128	Reserved
0001 0101b	n/a	21/128	21/128	Reserved
0001 0110b	n/a	22/128	22/128	Reserved
0001 0111b	n/a	23/128	23/128	Reserved
0001 1000b	n/a	24/128	24/128	Reserved
0001 1001b	n/a	25/128	25/128	Reserved
0001 1010b	n/a	26/128	26/128	Reserved
0001 1011b	n/a	27/128	27/128	Reserved
0001 1100b	n/a	28/128	28/128	Reserved
0001 1101b	n/a	29/128	29/128	Reserved
0001 1110b	n/a	30/128	30/128	Reserved
0001 1111b	n/a	31/128	31/128	Reserved
0010 0000b	n/a	32/128	32/128	Reserved
0010 0001b ～ 0011 1111b	n/a	33/128 ～ 63/128	33/128 ～ 63/128	Reserved
0100 0000b	n/a	64/128	64/128	Reserved
0100 0001b ～ 0111 1111b	n/a	65/128 ～ 127/128	65/128 ～ 127/128	Reserved
1000 0000b	n/a	128/128	128/128	Reserved

bits 7-0 View Resizer Horizontal Scaling Rate bits [7:0]
These bits determine the view resizer horizontal scaling rate when REG[2440h] bit 2 = 1b. When REG[2440h] bit 2 = 0b, these bits specify both the horizontal and the vertical scaling rate. Not all scaling rates are available for all scaling modes (see REG[244Eh]).

Table 10-84: View Resizer Horizontal Scaling Rate Selection

REG[244Ch] bits 7-0	**View Resizer Horizontal Scaling Rate**			
	REG[244Eh] bits 1-0 = 00b	**REG[244Eh] bits 1-0 = 01b**	**REG[244Eh] bits 1-0 = 10b**	**REG[244Eh] bits 1-0 = 11b**
0000 0000b	Reserved	Reserved	Reserved	Reserved
0000 0001b	n/a	1/128	1/128	Reserved
0000 0010b	n/a	2/128	2/128	Reserved
0000 0011b	n/a	3/128	Reserved	Reserved
0000 0100b	n/a	4/128	4/128	Reserved
0000 0101b	n/a	5/128	Reserved	Reserved
0000 0110b	n/a	6/128	Reserved	Reserved
0000 0111b	n/a	7/128	Reserved	Reserved
0000 1000b	n/a	8/128	8/128	Reserved
0000 1001b	n/a	9/128	Reserved	Reserved
0000 1010b	n/a	10/128	Reserved	Reserved
0000 1011b	n/a	11/128	Reserved	Reserved
0000 1100b	n/a	12/128	Reserved	Reserved
0000 1101b	n/a	13/128	Reserved	Reserved
0000 1110b	n/a	14/128	Reserved	Reserved
0000 1111b	n/a	15/128	Reserved	Reserved
0001 0000b	n/a	16/128	16/128	Reserved
0001 0001b	n/a	17/128	Reserved	Reserved
0001 0010b	n/a	18/128	Reserved	Reserved
0001 0011b	n/a	19/128	Reserved	Reserved
0001 0100b	n/a	20/128	Reserved	Reserved
0001 0101b	n/a	21/128	Reserved	Reserved
0001 0110b	n/a	22/128	Reserved	Reserved
0001 0111b	n/a	23/128	Reserved	Reserved
0001 1000b	n/a	24/128	Reserved	Reserved
0001 1001b	n/a	25/128	Reserved	Reserved
0001 1010b	n/a	26/128	Reserved	Reserved
0001 1011b	n/a	27/128	Reserved	Reserved
0001 1100b	n/a	28/128	Reserved	Reserved
0001 1101b	n/a	29/128	Reserved	Reserved
0001 1110b	n/a	30/128	Reserved	Reserved
0001 1111b	n/a	31/128	Reserved	Reserved
0010 0000b	n/a	32/128	32/128	Reserved
0010 0001b ～ 0011 1111b	n/a	33/128 ～ 63/128	Reserved	Reserved
0100 0000b	n/a	64/128	64/128	Reserved
0100 0001b ～ 0111 1111b	n/a	65/128 ～ 127/128	Reserved	Reserved
1000 0000b	n/a	128/128	128/128	Reserved

Note

The resizer output size, after trimming and scaling, must be an even number of pixels.

REG[244Eh] View Resizer Scaling Mode Register
Default = 0000h Read/Write

n/a							
15	14	13	12	11	10	9	8
n/a				Reserved		View Resizer Scaling Mode bits 1-0	
7	6	5	4	3	2	1	0

bits 3-2 Reserved
The default value for these bits is 00b.

bits 1-0 View Resizer Scaling Mode bits [1:0]
These bits determine the view resizer scaling mode. Not all scaling modes are available for all scaling rates. Before selecting a scaling mode, set the View Resizer Vertical Scaling Rate bits (REG[244Ch] bits 15-8) and/or the View Resizer Horizontal Scaling Rate bits (REG[244Ch] bits 7-0) to a valid scaling rate. Enabling a scaling mode with an unsupported scaling rate (reserved or n/a) may turn off the view resizer..

Table 10-85: View Resizer Scaling Mode Selection

REG[244Eh] bits 1-0	View Resizer Scaling Mode
00b	No resizer scaling
01b	V/H Reduction
10b	V: Reduction, H: Average
11b	Reserved

Note
The resizer output size, after trimming and scaling, must be an even number of pixels.

Capture (Encode) Resizer Registers

REG[2460h] Capture Resizer Control Register
Default = 0000h Read/Write

n/a							
15	14	13	12	11	10	9	8
Capture Resizer Software Reset (WO)	n/a				Capture Resizer Independent Horizontal/Vertical Scaling Enable	Capture Resizer Register Update VSYNC Enable	Capture Resizer Enable
7	6	5	4	3	2	1	0

bit 7 Capture Resizer Software Reset (Write Only)
This bit performs a software reset of the capture resizer logic.
Writing a 0b to this bit has no hardware effect.
Writing a 1b to this bit while the resizers are activated (REG[2440h] bit 0 = 1b or REG[2460h] bit 0 = 1b) performs a software reset of the capture resizer.

bit 2 Capture Resizer Independent Horizontal/Vertical Scaling Enable
This bit determines whether the capture resizer uses the same scaling rate for the horizontal and vertical directions, or uses independent scaling rates.
When this bit = 0b, the horizontal and vertical scaling rates are the same. Both horizontal and vertical scaling rates are controlled by REG[246Ch] bits 7-0.
When this bit = 1b, the horizontal and vertical scaling rates can be selected independently. Horizontal scaling rate is controlled by REG[246Ch] bits 7-0 and vertical scaling rate is controlled by REG[246Ch] bits 15-8.

bit 1 Capture Resizer Register Update VSYNC Enable
This bit determines whether the capture resizer logic uses new values programmed to the capture resizer registers immediately, or only once the next VSYNC occurs.
When this bit = 0b, the capture resizer uses the new register values immediately.
When this bit = 1b, the capture resizer uses the previous register values until the next VSYNC occurs when it uses the new register values.

bit 0 Capture Resizer Enable
This bit controls the capture resizer.
When this bit = 0b, the capture resizer is disabled.
When this bit = 1b, the capture resizer is enabled.

Note

1. The Capture Resizer and the View Resizer (REG[2440h] bit 0) must not be enabled at the same time.
2. When this bit and the View Resizer Enable bit (REG[2440h] bit 0) are both set to 0b, the clock to the resizer block is automatically stopped.

REG[2464h] Capture Resizer Start X Position Register Default = 0000h							Read/Write
n/a					Capture Resizer Start X Position bits 10-0		
15	14	13	12	11	10	9	8
Capture Resizer Start X Position bits 7-0							
7	6	5	4	3	2	1	0

bits 10-0 Capture Resizer Start X Position bits [10:0]
These bits determine the X start position for the Capture Resizer. These bits must be programmed according to the restrictions in Section 14, “Resizers” on page 393.

The following image size limitations must be observed when the YUV Capture function is used.

Table 10-86: Capture Resizer Limitations for YUV Capture

YUV Format	Minimum Horizontal Resolution	Minimum Vertical Resolution	Minimum Size
YUV 4:2:2	multiples of 2 pixels	multiples of 1 line	16 pixels/8 lines
YUV 4:2:0	multiples of 2 pixels	multiples of 2 lines	16 pixels/16 lines

REG[2466h] Capture Resizer Start Y Position Register
Default = 0000h Read/Write

n/a					Capture Resizer Start Y Position bits 10-8		
15	14	13	12	11	10	9	8
Capture Resizer Start Y Position bits 7-0							
7	6	5	4	3	2	1	0

bits 10-0 Capture Resizer Start Y Position bits [10:0]
These bits determine the Y start position for the Capture Resizer. These bits must be programmed according to the restrictions in Section 14, “Resizers” on page 393.

REG[2468h] Capture Resizer End X Position Register
Default = 027Fh Read/Write

n/a					Capture Resizer End X Position bits 10-8		
15	14	13	12	11	10	9	8
Capture Resizer End X Position bits 7-0							
7	6	5	4	3	2	1	0

bits 10-0 Capture Resizer End X Position bits [10:0]
These bits determine the X end position for the Capture Resizer. These bits must be programmed according to the restrictions in Section 14, “Resizers” on page 393.

REG[246Ah] Capture Resizer End Y Position Register
Default = 01DFh Read/Write

n/a					Capture Resizer End Y Position bits 10-8		
15	14	13	12	11	10	9	8
Capture Resizer End Y Position bits 7-0							
7	6	5	4	3	2	1	0

bits 10-0 Capture Resizer End Y Position bits [10:0]
These bits determine the Y end position for the Capture Resizer. These bits must be programmed according to the restrictions in Section 14, “Resizers” on page 393.

REG[246Ch] Capture Resizer Scaling Rate Register Default = 8080h							Read/Write
Capture Resizer Vertical Scaling Rate bits 7-0							
15	14	13	12	11	10	9	8
Capture Resizer Horizontal Scaling Rate bits 7-0							
7	6	5	4	3	2	1	0

bits 15-8 Capture Resizer Vertical Scaling Rate bits [7:0]
These bits determine the capture resizer vertical scaling rate when REG[2460h] bit 2 = 1b. Not all scaling rates are available for all scaling modes (see REG[246Eh]).

Table 10-87: Capture Resizer Vertical Scaling Rate Selection

REG[246Ch] bits 15-8	**Capture Resizer Vertical Scaling Rate**			
	REG[246Eh] bits 1-0 = 00b	**REG[246Eh] bits 1-0 = 01b**	**REG[246Eh] bits 1-0 = 10b**	**REG[246Eh] bits 1-0 = 11b**
0000 0000b	Reserved	Reserved	Reserved	Reserved
0000 0001b	n/a	1/128	1/128	Reserved
0000 0010b	n/a	2/128	2/128	Reserved
0000 0011b	n/a	3/128	3/128	Reserved
0000 0100b	n/a	4/128	4/128	Reserved
0000 0101b	n/a	5/128	5/128	Reserved
0000 0110b	n/a	6/128	6/128	Reserved
0000 0111b	n/a	7/128	7/128	Reserved
0000 1000b	n/a	8/128	8/128	Reserved
0000 1001b	n/a	9/128	9/128	Reserved
0000 1010b	n/a	10/128	10/128	Reserved
0000 1011b	n/a	11/128	11/128	Reserved
0000 1100b	n/a	12/128	12/128	Reserved
0000 1101b	n/a	13/128	13/128	Reserved
0000 1110b	n/a	14/128	14/128	Reserved
0000 1111b	n/a	15/128	15/128	Reserved
0001 0000b	n/a	16/128	16/128	Reserved
0001 0001b	n/a	17/128	17/128	Reserved
0001 0010b	n/a	18/128	18/128	Reserved
0001 0011b	n/a	19/128	19/128	Reserved
0001 0100b	n/a	20/128	20/128	Reserved
0001 0101b	n/a	21/128	21/128	Reserved
0001 0110b	n/a	22/128	22/128	Reserved
0001 0111b	n/a	23/128	23/128	Reserved
0001 1000b	n/a	24/128	24/128	Reserved
0001 1001b	n/a	25/128	25/128	Reserved
0001 1010b	n/a	26/128	26/128	Reserved
0001 1011b	n/a	27/128	27/128	Reserved
0001 1100b	n/a	28/128	28/128	Reserved
0001 1101b	n/a	29/128	29/128	Reserved
0001 1110b	n/a	30/128	30/128	Reserved
0001 1111b	n/a	31/128	31/128	Reserved
0010 0000b	n/a	32/128	32/128	Reserved
0010 0001b 〜 0011 1111b	n/a	33/128 〜 63/128	33/128 〜 63/128	Reserved
0100 0000b	n/a	64/128	64/128	Reserved
0100 0001b 〜 0111 1111b	n/a	65/128 〜 127/128	65/128 〜 127/128	Reserved
1000 0000b	n/a	128/128	128/128	Reserved

bits 7-0 Capture Resizer Horizontal Scaling Rate bits [7:0]
These bits determine the capture resizer horizontal scaling rate when REG[2460h] bit 2 = 1b. When REG[2460h] bit 2 = 0b, these bits specify both the horizontal and the vertical scaling rate. Not all scaling rates are available for all scaling modes (see REG[246Eh]).

Table 10-88: Capture Resizer Horizontal Scaling Rate Selection

REG[246Ch] bits 7-0	**Capture Resizer Horizontal Scaling Rate**			
	REG[246Eh] bits 1-0 = 00b	**REG[246Eh] bits 1-0 = 01b**	**REG[246Eh] bits 1-0 = 10b**	**REG[246Eh] bits 1-0 = 11b**
0000 0000b	Reserved	Reserved	Reserved	Reserved
0000 0001b	n/a	1/128	1/128	Reserved
0000 0010b	n/a	2/128	2/128	Reserved
0000 0011b	n/a	3/128	Reserved	Reserved
0000 0100b	n/a	4/128	4/128	Reserved
0000 0101b	n/a	5/128	Reserved	Reserved
0000 0110b	n/a	6/128	Reserved	Reserved
0000 0111b	n/a	7/128	Reserved	Reserved
0000 1000b	n/a	8/128	8/128	Reserved
0000 1001b	n/a	9/128	Reserved	Reserved
0000 1010b	n/a	10/128	Reserved	Reserved
0000 1011b	n/a	11/128	Reserved	Reserved
0000 1100b	n/a	12/128	Reserved	Reserved
0000 1101b	n/a	13/128	Reserved	Reserved
0000 1110b	n/a	14/128	Reserved	Reserved
0000 1111b	n/a	15/128	Reserved	Reserved
0001 0000b	n/a	16/128	16/128	Reserved
0001 0001b	n/a	17/128	Reserved	Reserved
0001 0010b	n/a	18/128	Reserved	Reserved
0001 0011b	n/a	19/128	Reserved	Reserved
0001 0100b	n/a	20/128	Reserved	Reserved
0001 0101b	n/a	21/128	Reserved	Reserved
0001 0110b	n/a	22/128	Reserved	Reserved
0001 0111b	n/a	23/128	Reserved	Reserved
0001 1000b	n/a	24/128	Reserved	Reserved
0001 1001b	n/a	25/128	Reserved	Reserved
0001 1010b	n/a	26/128	Reserved	Reserved
0001 1011b	n/a	27/128	Reserved	Reserved
0001 1100b	n/a	28/128	Reserved	Reserved
0001 1101b	n/a	29/128	Reserved	Reserved
0001 1110b	n/a	30/128	Reserved	Reserved
0001 1111b	n/a	31/128	Reserved	Reserved
0010 0000b	n/a	32/128	32/128	Reserved
0010 0001b ～0011 1111b	n/a	33/128 ～ 63/128	Reserved	Reserved
0100 0000b	n/a	64/128	64/128	Reserved
0100 0001b ～0111 1111b	n/a	65/128 ～ 127/128	Reserved	Reserved
1000 0000b	n/a	128/128	128/128	Reserved

REG[246Eh] Capture Resizer Scaling Mode Register
Default = 0000h Read/Write

n/a							
15	14	13	12	11	10	9	8
n/a				Reserved		Capture Resizer Scaling Mode bits 1-0	
7	6	5	4	3	2	1	0

bits 3-2 Reserved
The default value for these bits is 00b.

bits 1-0 Capture Resizer Scaling Mode bits [1:0]
These bits determine the capture resizer scaling mode. Not all scaling rates are available for all scaling modes. Before selecting a scaling mode, set the Capture Resizer Vertical Scaling Rate bits (REG[246Ch] bits 15-8) and/or the Capture Resizer Horizontal Scaling Rate bits (REG[246Ch] bits 7-0) to a valid scaling rate. Enabling a scaling mode with an unsupported scaling rate (reserved or n/a) may turn off the capture resizer.

Table 10-89: Capture Resizer Scaling Mode Selection

REG[246Eh] bits 1-0	Capture Resizer Scaling Mode
00b	No resizer scaling
01b	V/H Reduction
10b	V: Reduction, H: Average
11b	Reserved

10.4.13 YUV Capture Module Registers

REG[2800h] YUV Capture Control Register
Default = 0000h Read/Write

Reserved							
15	14	13	12	11	10	9	8
YUV Capture Module Software Reset (WO)	Reserved			YUV Capture Data Flow Control bits 2-0			YUV Capture Module Enable
7	6	5	4	3	2	1	0

bits 15-8 Reserved
The default value for these bits is 0000_0000b.

bit 7 YUV Capture Module Software Reset (Write Only)
This bit performs a software reset of the YUV Capture module logic. The YUV Capture module should be reset using this bit before each YUV Capture operation. This bit resets the following registers: REG[2802h] except for bits 10-8, REG[2804h] except for bits 10-8, REG[2822h], and REG[2828h]~ REG[282Ch]. All other YUV Capture Module registers retain their previous values. This bit automatically resets to 0b once the YUV Capture Module software reset completes.
Writing a 0b to this bit has no hardware effect.
Writing a 1b to this bit initiates a software reset of the YUV Capture Module.

bits 6-4 Reserved
The default value for these bits is 000b.

bits 3-1 YUV Capture Data Flow Control bits [2:0]
These bits control the data flow through the YUV Capture module.

Table 10-90: YUV Capture Data Mode Selection

REG[2800h] bits 3-1	Data Flow Mode
000b	Reserved
001b	**YUV Data Input from Memory (YUV 4:2:2)** YUV 4:2:2 format image data from the memory is sent through the YRC (YUV to RGB Converter) and then written back to memory. (see Note 1)
010b	Reserved
011b	**YUV Data Output to Memory (YUV 4:2:2)** YUV 4:2:2 format image data is received from the camera interface and written to memory.
100b ~ 110b	Reserved
111b	**YUV Data Output to Memory (YUV 4:2:0)** YUV 4:2:0 format image data is received from the camera interface and written to memory. (see Note 2)

Note

1. When these bits = 001b, REG[2440h] bit 1 must be set to 0b.
2. When performing YUV 4:2:0 capture (REG[2800h] bits 3-1 = 111b), the capture resizer horizontal size (or width) must be a multiple of 4.

bit 0 YUV Capture Module Enable
This bit controls the YUV Capture module.
When this bit = 0b, the capture module is disabled and the clock source is disabled.
When this bit = 1b, the capture module is enabled and the clock source is supplied.

REG[2802h] YUV Capture Status Flag Register
Default = 8080h Read/Write

Reserved		YUV Capture FIFO Threshold Status bits 1-0 (RO)		Reserved	YUV Capture FIFO Threshold Trigger Flag	YUV Capture FIFO Full Flag	YUV Capture FIFO Empty Flag
15	14	13	12	11	10	9	8
Reserved							
7	6	5	4	3	2	1	0

bits 15-14 Reserved
The default value for these bits is 10b.

bits 13-12 YUV Capture FIFO Threshold Status bits [1:0] (Read Only)
These bits indicate how much data is currently in the YUV Capture FIFO. For information on setting the YUV Capture FIFO size, see the YUV Capture FIFO Size register (REG[2824h]).

Table 10-91: YUV Capture FIFO Threshold Status

REG[2802h] bits 13-12	YUV Capture FIFO Threshold Status
00b	No data exists (same as empty)
01b	3 or more dwords exist in the YUV Capture FIFO + Output FIFO
10b	1/4 or more of the specified YUV Capture FIFO size data exists
11b	1/2 or more of the specified YUV Capture FIFO size data exists

bit 11 Reserved
The default value for this bit is 0b.

bit 10 YUV Capture FIFO Threshold Trigger Flag
This flag is asserted when the amount of data in the YUV Capture FIFO meets the condition specified by the YUV Capture FIFO Trigger Threshold bits (REG[2820h] bits 5-4).
This flag is masked by the YUV Capture FIFO Threshold Trigger Interrupt Enable bit and is only available when REG[2806h] bit 10 = 1b.
When this bit = 0b, the amount of data in the YUV Capture FIFO is less than the YUV Capture FIFO Trigger Threshold.
When this bit = 1b, the amount of data in the YUV Capture FIFO has reached the YUV Capture FIFO Trigger Threshold.

To clear this bit, write a 1b to this bit.

bit 9 YUV Capture FIFO Full Flag
This flag is asserted when both the YUV Capture FIFO and the YUV Capture Input FIFO are full. For details on the YUV Capture FIFO structure, see Figure 10-6: “YUV Capture FIFO Structure” on page 314. This flag is masked by the YUV Capture FIFO Full Interrupt Enable bit and is only available when REG[2806h] bit 9 = 1b.
When this bit = 0b, the YUV Capture FIFO and YUV Capture Input FIFO are not full.
When this bit = 1b, the YUV Capture FIFO and YUV Capture Input FIFO are full.

To clear this bit, write a 1b to this bit.

bit 8 YUV Capture FIFO Empty Flag
This flag is asserted when the YUV Capture Output FIFO contains less than 3 dwords of data. For details on the YUV Capture FIFO structure, see Figure 10-6: “YUV Capture FIFO Structure” on page 314. This flag is masked by the YUV Capture FIFO Empty Interrupt Enable bit and is only available when REG[2806h] bit 8 = 1b.
When this bit = 0b, the YUV Capture Output FIFO contains 3 or more dwords of data.
When this bit = 1b, the YUV Capture Output FIFO contains less than 3 dwords of data.

To clear this bit, write a 1b to this bit.

bits 7-0 Reserved
The default value for these bits is 1000_0000b.

REG[2804h] YUV Capture Raw Status Flag Register
Default = 8080h Read Only

Reserved		Raw YUV Capture FIFO Threshold Status bits 1-0		Reserved	Raw YUV Capture FIFO Threshold Trigger Flag	Raw YUV Capture FIFO Full Flag	Raw YUV Capture FIFO Empty Flag
15	14	13	12	11	10	9	8
Reserved							
7	6	5	4	3	2	1	0

bits 15-14 Reserved
The default value for these bits is 10b.

bits 13-12 Raw YUV Capture FIFO Threshold Status bits [1:0] (Read Only)
These bits indicate how much data is currently in the YUV Capture FIFO. For information on setting the YUV Capture FIFO size, see the YUV Capture FIFO Size register (REG[2824h]).

Table 10-92: Raw YUV Capture FIFO Threshold Status

REG[2804h] bits 13-12	Raw YUV Capture FIFO Threshold Status
00b	No data exists (same as empty)
01b	3 or more dwords exist in the YUV Capture FIFO + Output FIFO
10b	1/4 or more of the specified YUV Capture FIFO size data exists
11b	1/2 or more of the specified YUV Capture FIFO size data exists

bit 11 Reserved
The default value for this bit is 0b.

bit 10 Raw YUV Capture FIFO Threshold Trigger Flag (Read Only)
This flag is asserted when the amount of data in the YUV Capture FIFO meets the condition specified by the YUV Capture FIFO Trigger Threshold bits (REG[2820h] bits 5-4). This flag is not masked by the YUV Capture FIFO Threshold Trigger Interrupt Enable bit, REG[2806h] bit 10.
When this bit = 0b, the amount of data in the YUV Capture FIFO is less than the YUV Capture FIFO Trigger Threshold.
When this bit = 1b, the amount of data in the YUV Capture FIFO has reached the YUV Capture FIFO Trigger Threshold.

To clear this bit, write a 1b to the YUV Capture FIFO Threshold Trigger Flag (REG[2802h] bit 10).

bit 9 Raw YUV Capture FIFO Full Flag (Read Only)
This flag is asserted when both the YUV Capture FIFO and the YUV Capture Input FIFO are full. For details on the YUV Capture FIFO structure, see Figure 10-6: “YUV Capture FIFO Structure” on page 314. This flag is not affected by the YUV Capture FIFO Full Interrupt Enable bit, REG[2806h] bit 9.
When this bit = 0b, the YUV Capture FIFO and YUV Capture Input FIFO are not full.
When this bit = 1b, the YUV Capture FIFO and YUV Capture Input FIFO are full.

To clear this flag, write a 1b to the YUV Capture FIFO Full Flag (REG[2802h] bit 9) when the YUV Capture FIFO is no longer full or after a YUV Capture module software reset, REG[2800h] bit 7 = 1b.

bit 8 Raw YUV Capture FIFO Empty Flag (Read Only)
This flag is asserted when the YUV Capture Output FIFO contains less than 2 dwords of data. For details on the YUV Capture FIFO structure, see Figure 10-6: “YUV Capture FIFO Structure” on page 314. This flag is not affected by the YUV Capture FIFO Empty Interrupt Enable bit, REG[2806h] bit 8.
When this bit = 0b, the YUV Capture Output FIFO contains 2 or more dwords of data.
When this bit = 1b, the YUV Capture Output FIFO contains less than 2 dwords of data.

To clear this flag, write a 1b to the YUV Capture FIFO Empty Flag (REG[2802h] bit 8) when the YUV Capture FIFO is no longer empty or after a YUV Capture module software reset, REG[2800h] bit 7 = 1b.

Note

This bit is not affected by the YUV Capture FIFO Clear bit, REG[2820h] bit 2.

bits 7-0 Reserved
The default value for these bits is 1000_0000b.

REG[2806h] YUV Capture Interrupt Control Register Default = 0000h							Read/Write
Reserved					YUV Capture FIFO Threshold Trigger Interrupt Enable	YUV Capture FIFO Full Interrupt Enable	YUV Capture FIFO Empty Interrupt Enable
15	14	13	12	11	10	9	8
Reserved							
7	6	5	4	3	2	1	0

bits 15-12 — Reserved
The default value for these bits is 0_0000b.

bit 10 — YUV Capture FIFO Threshold Trigger Interrupt Enable
This bit controls the YUV Capture FIFO Threshold Trigger interrupt. The status of this interrupt is indicated by the YUV Capture FIFO Threshold Trigger Flag bit, REG[2802h] bit 10.
When this bit = 0b, the interrupt is disabled.
When this bit = 1b, the interrupt is enabled.

bit 9 — YUV Capture FIFO Full Interrupt Enable
This bit controls the YUV Capture FIFO Full interrupt. The status of this interrupt is indicated by the YUV Capture FIFO Full Flag bit, REG[2802h] bit 9.
When this bit = 0b, the interrupt is disabled.
When this bit = 1b, the interrupt is enabled.

bit 8 — YUV Capture FIFO Empty Interrupt Enable
This bit controls the YUV Capture FIFO Empty interrupt. The status of this interrupt is indicated by the YUV Capture FIFO Empty Flag bit, REG[2802h] bit 8.
When this bit = 0b, the interrupt is disabled.
When this bit = 1b, the interrupt is enabled.

bits 7-0 — Reserved
The default value for these bits is 0000_0000b.

REG[2808h] is Reserved

This register is Reserved and should not be written.

REG[280Ah] YUV Capture Start/Stop Control Register Default = 0000h							Write Only
n/a							
15	14	13	12	11	10	9	8
n/a							YUV Capture Start/Stop Control
7	6	5	4	3	2	1	0

bit 0 YUV Capture Start/Stop Control (Write Only)
This bit controls when a YUV data capture is started or stopped.
Writing a 0b to this bit stops YUV data capturing at the end of the current frame.
Writing a 1b to this bit starts YUV data capturing at the beginning of the next frame.

Note
For continuous capture mode, one out of every two frames are captured.

REG[280Ch] through REG[280Eh] are Reserved

These registers are Reserved and should not be written.

10.4.14 YUV Capture FIFO Registers

REG[2820h] YUV Capture FIFO Control Register Default = 0000h							Read/Write
Reserved							
15	14	13	12	11	10	9	8
Reserved		YUV Capture FIFO Trigger Threshold bits 1-0		Reserved	YUV Capture FIFO Clear (WO)	n/a	YUV Capture FIFO Direction (RO)
7	6	5	4	3	2	1	0

bits 15-6 Reserved
The default value for these bits is 00_0000_0000b.

bits 5-4 YUV Capture FIFO Trigger Threshold bits [1:0]
These bits set the YUV Capture FIFO Threshold Trigger Flag (REG[2802h] bit 10) when the specified conditions are met.

Table 10-93: YUV Capture FIFO Trigger Threshold Selection

REG[2820h] bits 5-4	YUV Capture FIFO Trigger Threshold
00b	Never trigger
01b	Reserved
10b	Trigger when the YUV Capture FIFO contains more than 1/4 of the specified YUV Capture FIFO size (REG[2824h] bits 7-0), but less than 1/2 of the YUV capture FIFO size.
11b	Trigger when the YUV Capture FIFO contains more than 1/2 of the specified YUV Capture FIFO size (REG[2824h] bits 7-0)

bit 3 Reserved
The default value for this bit is 0b.

bit 2 YUV Capture FIFO Clear (Write Only)
This bit is used to clear the YUV Capture FIFO. It is recommended that the YUV Capture module should also be reset (REG[2800h] bit 7 = 1b) when the YUV Capture FIFO is cleared.
Writing a 0b to this bit has no hardware effect.
Writing a 1b to this bit clears the YUV Capture FIFO, YUV Capture FIFO Read/Write Pointer registers (REG[282Ah] ~ REG[282Ch]), and the YUV Capture FIFO Valid Data Size registers (REG[2828h]).

The following procedure should be used to clear the YUV Capture FIFO.

1. Clear the YUV Capture FIFO, REG[2820h] bit 2 = 1b.
2. Perform 2 dummy reads from REG[2826h] to ensure that the YUV Capture FIFO is empty.
3. Reset the YUV Capture module, REG[2800h] bit 7 = 1b.

Note
Clearing the Capture FIFO using this bit has no effect on the Raw YUV Capture FIFO Empty Flag, REG[2804h] bit 8.

bit 0 YUV Capture FIFO Direction (Read Only)
This bit indicates the configuration of the YUV Capture FIFO.
When this bit = 0b, the YUV Capture FIFO is configured to receive. For example, receive mode is used when YUV data is captured from the camera and stored in memory.

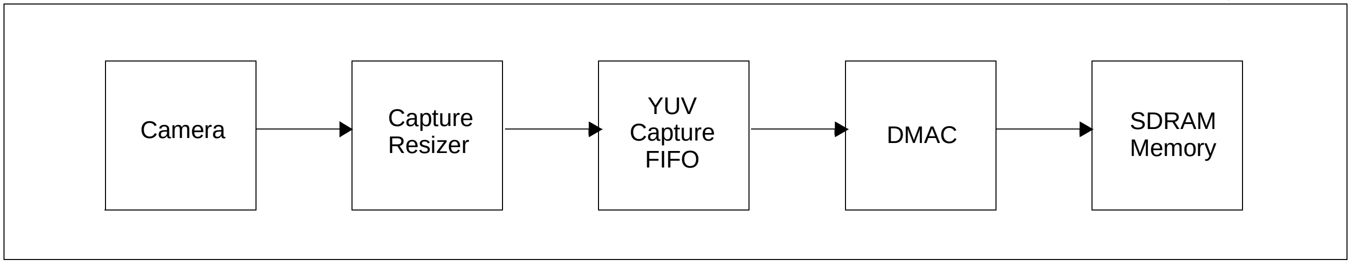


Figure 10-4: YUV Capture FIFO Receive Mode Example

When this bit = 1b, the YUV Capture FIFO is configured to transmit. For example, transmit mode is used when YUV data from the Host is written to the memory for eventual conversion to RGB using the YRC and display on the LCD panel.

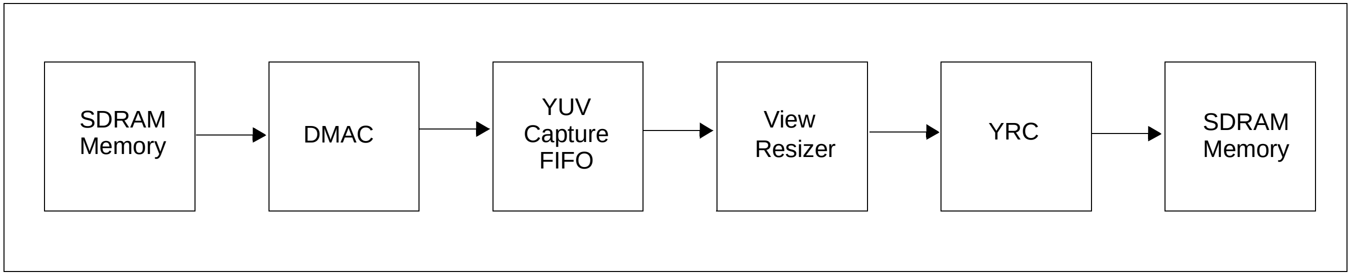


Figure 10-5: YUV Capture FIFO Transmit Mode Example

REG[2822h] YUV Capture FIFO Status Register Default = 8001h							Read Only
Reserved							
15	14	13	12	11	10	9	8
Reserved				YUV Capture FIFO Threshold Status bits 1-0		YUV Capture FIFO Full Status	YUV Capture FIFO Empty Status
7	6	5	4	3	2	1	0

bits 15-4 Reserved
The default value for these bits is 1000_0000_0000b.

bits 3-2 YUV Capture FIFO Threshold Status bits [1:0] (Read Only)
These bits indicate how much data is currently in the YUV Capture FIFO. For information on setting the YUV Capture FIFO size, see the YUV Capture FIFO Size register (REG[2824h]).

Table 10-94: YUV Capture FIFO Threshold Status

REG[2822h] bits 3-2	YUV Capture FIFO Threshold Status
00b	No data exists (same as empty)
01b	3 or more dwords exist in the YUV Capture FIFO + Output FIFO
10b	More than 1/4 of the specified YUV Capture FIFO size data exists
11b	More than 1/2 of the specified YUV Capture FIFO size data exists

bit 1 YUV Capture FIFO Full Status (Read Only)
This bit indicates whether the YUV Capture FIFO is full.
When this bit = 0b, the YUV Capture FIFO is not full.
When this bit = 1b, the YUV Capture FIFO is full.

bit 0 YUV Capture FIFO Empty Status (Read Only)
This bit indicates that the YUV Capture FIFO is empty.
When this bit = 0b, the YUV Capture FIFO is not empty.
When this bit = 1b, the YUV Capture FIFO is empty.

REG[2824h] YUV Capture FIFO Size Register
Default = 003Fh Read/Write

15	14	13	12	11	10	9	8
n/.a							

7	6	5	4	3	2	1	0
YUV Capture FIFO Size bits 7-0							

bits 7-0 YUV Capture FIFO Size bits [7:0]
These bits determine the YUV Capture FIFO size, in 4 byte units. The maximum size of the YUV Capture FIFO is 256 bytes plus 24 bytes which are automatically allocated for small input (2 dword) and output (4 dword) FIFOs. It is recommended to leave the YUV Capture FIFO at the default setting of 256 bytes. If the YUV Capture FIFO size must be changed, the size is calculated using the following formula.

YUV Capture FIFO size = (REG[2824h] bits 7-0 + 7) x 4 bytes

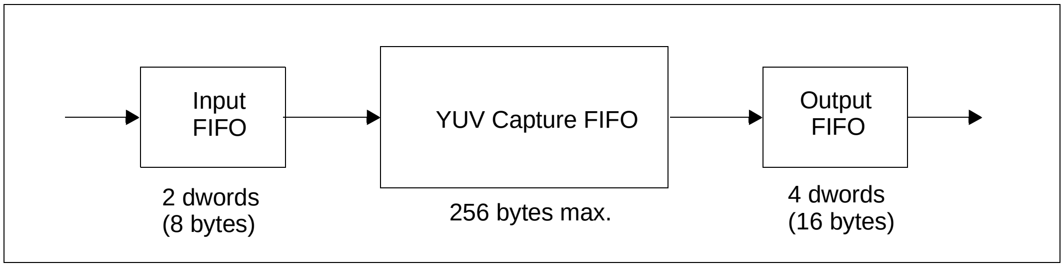


Figure 10-6: YUV Capture FIFO Structure

REG[2826h] YUV Capture FIFO Read/Write Port Register
Default = Not Applicable Read/Write

15	14	13	12	11	10	9	8
YUV Capture FIFO Read/Write Port bits 15-8							

7	6	5	4	3	2	1	0
YUV Capture FIFO Read/Write Port bits 7-0							

bits 15-0 YUV Capture FIFO Read/Write Port bits [15:0]
These bits are the access port for the YUV Capture FIFO.
When YUV data is output to the memory from the capture interface (REG[2800h] bits 3-1 = 011b or 111b), these bits are used as the YUV Capture FIFO read data port by the DMA controller. When YUV data in memory is written to the display memory as RGB data (REG[2800h] bits 3-1 = 001b), these bits are used as the YUV Capture FIFO write data port by the DMA controller.

Note

This register can only be accessed by the DMA controller (DMAC) at DMA address 3000_0000h.

REG[2828h] YUV Capture FIFO Valid Data Size Register Default = 0000h							Read Only
YUV Capture FIFO Valid Data Size bits 15-8							
15	14	13	12	11	10	9	8
YUV Capture FIFO Valid Data Size bits 7-0							
7	6	5	4	3	2	1	0

bits 15-0 YUV Capture FIFO Valid Data Size bits [15:0] (Read Only)
These bits, in conjunction with REG[282Eh] bits 9-8, indicate the valid data size that can be read from the YUV Capture FIFO, in 32-bit (4 byte) units.

REG[282Ah] through REG[282Ch] are Reserved

These registers are Reserved and should not be written.

REG[282Eh] YUV Capture FIFO Extend Register Default = 0000h							Read Only
n/a						YUV Capture FIFO Valid Data Size bits 17-16	
15	14	13	12	11	10	9	8
n/a		Reserved		n/a		Reserved	
15	14	5	4	15	14	1	0

bits 9-8 YUV Capture FIFO Valid Data Size bits [17:16] (Read Only)
These bits extend the YUV Capture FIFO Valid Data Size (REG[2828h]) to 18-bits.

bits 5-4 Reserved
The default value for these bits is 00b.

bits 1-0 Reserved
The default value for these bits is 00b.

REG[2830h] through REG[2870h] are Reserved

These registers are Reserved and should not be written.

REG[2872h] YUV Horizontal Size Register Default = 0000h							Write Only
n/a					YUV Horizontal Size bits 10-8		
15	14	13	12	11	10	9	8
YUV Horizontal Size bits 7-0							
7	6	5	4	3	2	1	0

bits 10-0 YUV Horizontal Size bits [10:0]
These bits specify the horizontal (or X) size of the YUV image in memory to be displayed (YUV Display mode, REG[2800h] bits 3-1 = 001b). These bits are used to provide pseudo Vsync/Hsync signals for the View Resizer. These bits have no effect for YUV Capture modes, REG[2800h] bits 3-1 = 011b or 111b.
YUV Horizontal Size = REG[2872h] bits 10-0 + 1

REG[2874h] YUV Vertical Size Register Default = 0000h							Write Only
n/a					YUV Vertical Size bits 10-8		
15	14	13	12	11	10	9	8
YUV Vertical Size bits 7-0							
7	6	5	4	3	2	1	0

bits 10-0 YUV Vertical Size bits [10:0]
These bits specify the vertical (or Y) size of the YUV image in memory to be displayed (YUV Display mode, REG[2800h] bits 3-1 = 001b). These bits are used to provide pseudo Vsync/Hsync signals for the View Resizer. These bits have no effect for YUV Capture modes, REG[2800h] bits 3-1 = 011b or 111b.

YUV Vertical Size = REG[2874h] bits 10-0 + 1

Note

When performing YUV 4:2:2 display (REG[2800h] bits 3-1 = 001b) and the YUV horizontal size is less than or equal to 256 (REG[2872h] bits 10-0 ≤ 256), the YUV vertical size must also be less than or equal to 256 (REG[2874h] bits 10-0 ≤ 256).

REG[2876h] is Reserved

This register is Reserved and should not be written.

10.4.15 YRC Registers

REG[3000h] YRC Translate Mode Register Default = 0605h							Read/Write
YRC Bypass Mode Enable	YRC Rectangular Write Mode Enable	Reserved	YRC Frame Buffer Write Mode Select	YRC Output Bpp Select bits 1-0		YRC YUV Output Data Format Select	Reserved
15	14	13	12	11	10	9	8
YRC Software Reset	YRC UV Fix Select bits 1-0		YRC Input Data Type Select	n/a	YRC Transfer Mode bits 2-0		
7	6	5	4	3	2	1	0

bit 15 YRC Bypass Mode Enable
This bit controls YRC Bypass Mode and is only used for writes from the YRC (YUV to RGB Converter) to the external SDRAM memory. When YRC bypass mode is disabled, the YRC is used to convert YUV data from the camera interface into RGB format data which is stored in the SDRAM for display on the LCD panel. When YRC bypass mode is enabled, the YRC is bypassed which means that YUV data is sent directly to the SDRAM memory.
When this bit = 0b, YRC bypass mode is disabled. (default)
When this bit = 1b, YRC bypass mode is enabled.

Note

The YRC swaps the incoming byte data when it is disabled. To change the YUV data back to normal, set the YRC YUV Output Data Format Select bit (REG[3000h] bit 9) to 1b. Bypassing the YRC can be useful for cameras that are capable of outputting RGB data.

bit 14 YRC Rectangular Write Mode Enable
This bit controls YRC Rectangular Write Mode and is only used for writes from the YRC (YUV to RGB Converter) to the external SDRAM memory. YRC rectangular write mode should be used when the YRC source data is a different size than that of the destination area in memory. For example, rectangular write mode should be used if a camera image smaller than the display window will be placed on the main window.
When this bit = 0b, linear write mode is selected and data is written to the memory sequentially based on the YRC Write Start Address 0 registers (REG[3002h] ~ REG[3004h]), or if double buffer write mode is enabled (REG[3002h] ~ REG[3008h]).
When this bit = 1b, rectangular write mode is selected and data is written based on the YRC Write Start Address 0 registers (REG[3002h] ~ REG[3004h]), YRC Rectangular Pixel Width register (REG[3010h]), and the YRC Rectangular Line Address Offset register (REG[3012h]).

Note

YRC Rectangular Write Mode is only supported for single buffer write mode, REG[3000h] bit 12 = 0b.

bit 13 Reserved
The default value for this bit is 0b.

bit 12 YRC Frame Buffer Write Mode Select

This bit selects whether the YRC writes to SDRAM memory using one or two frame buffers and is only used for writes from the YRC (YUV to RGB Converter) to the external SDRAM memory. YRC double buffer write mode can be used in conjunction with the display double buffer function (see REG[0834h]) to prevent “tearing” of the camera image for fast moving images.

When this bit = 0b, single buffer write mode is selected. Single buffer write mode uses the YRC Write Start Address 0 registers (REG[3002h] ~ REG[3004h]).

When this bit = 1b, double buffer write mode is selected. Double buffer write mode uses both the YRC Write Start Address 0 registers (REG[3002h] ~ REG[3004h]) and the YRC Write Start Address 1 registers (REG[3006h] ~ REG[3008h]).

Note

YRC Rectangular Write Mode (REG[3000h] bit 14 = 1b) is not supported when double buffer write mode is selected, REG[3000h] bit 12 = 1b.

bits 11-10 YRC Output Bpp Select bits [1:0]

These bits are only used for writes from the YRC (YUV to RGB Converter) to the external SDRAM memory. These bits select the color depth, in bits-per-pixel (bpp), for YRC output when the YRC converts YUV data to RGB data.

Table 10-95: YRC Output Bpp Selection

REG[3000h] bits 11-10	YRC Output Bpp
00b	16 bpp
01b (default)	
10b	Reserved
11b	Reserved

bit 9 YRC YUV Output Data Format Select

This bit is only used for writes from the YRC (YUV to RGB Converter) to the external SDRAM memory. This bit selects the output data format of the YRC (YUV to RGB Converter) when it is bypassed, REG[3000h] bit 15 = 1b. This bit has no effect when the YRC is enabled, REG[3000h] bit 15 = 0b.

When this bit = 0b, VYUY format is selected. See Table 10-67:, “VYUY Output Data Format (REG[3000h] bit 9= 0b),” on page 206.

When this bit = 1b, YUYV format is selected. See Table 10-68:, “YUYV Output Data Format Select (REG[3000h] bit 9= 1b),” on page 206.

Table 10-96: VYUY Output Data Format (REG[3000h] bit 9 = 0b)

Cycle Count	**1**	**2**	**3**	**4**	**...**	**2n+1**	**2n+2**
D15	V_0^7	U_0^7	V_2^7	U_2^7	...	V_{2n}^7	U_{2n}^7
D14	V_0^6	U_0^6	V_2^6	U_2^6	...	V_{2n}^6	U_{2n}^6
D13	V_0^5	U_0^5	V_2^5	U_2^5	...	V_{2n}^5	U_{2n}^5
D12	V_0^4	U_0^4	V_2^4	U_2^4	...	V_{2n}^4	U_{2n}^4
D11	V_0^3	U_0^3	V_2^3	U_2^3	...	V_{2n}^3	U_{2n}^3
D10	V_0^2	U_0^2	V_2^2	U_2^2	...	V_{2n}^2	U_{2n}^2
D9	V_0^1	U_0^1	V_2^1	U_2^1	...	V_{2n}^1	U_{2n}^1
D8	V_0^0	U_0^0	V_2^0	U_2^0	...	V_{2n}^0	U_{2n}^0
D7	Y_1^7	Y_0^7	Y_3^7	Y_2^7	...	Y_{2n+1}^7	Y_{2n}^7
D6	Y_1^6	Y_0^6	Y_3^6	Y_2^6	...	Y_{2n+1}^6	Y_{2n}^6
D5	Y_1^5	Y_0^5	Y_3^5	Y_2^5	...	Y_{2n+1}^5	Y_{2n}^5
D4	Y_1^4	Y_0^4	Y_3^4	Y_2^4	...	Y_{2n+1}^4	Y_{2n}^4
D3	Y_1^3	Y_0^3	Y_3^3	Y_2^3	...	Y_{2n+1}^3	Y_{2n}^3
D2	Y_1^2	Y_0^2	Y_3^2	Y_2^2	...	Y_{2n+1}^2	Y_{2n}^2
D1	Y_1^1	Y_0^1	Y_3^1	Y_2^1	...	Y_{2n+1}^1	Y_{2n}^1
D0	Y_1^0	Y_0^0	Y_3^0	Y_2^0	...	Y_{2n+1}^0	Y_{2n}^0

Table 10-97: YUYV Output Data Format Select (REG[3000h] bit 9 = 1b)

Cycle Count	**1**	**2**	**3**	**4**	**...**	**2n+1**	**2n+2**
D15	Y_0^7	Y_1^7	Y_2^7	Y_3^7	...	Y_{2n}^7	Y_{2n+1}^7
D14	Y_0^6	Y_1^6	Y_2^6	Y_3^6	...	Y_{2n}^6	Y_{2n+1}^6
D13	Y_0^5	Y_1^5	Y_2^5	Y_3^5	...	Y_{2n}^5	Y_{2n+1}^5
D12	Y_0^4	Y_1^4	Y_2^4	Y_3^4	...	Y_{2n}^4	Y_{2n+1}^4
D11	Y_0^3	Y_1^3	Y_2^3	Y_3^3	...	Y_{2n}^3	Y_{2n+1}^3
D10	Y_0^2	Y_1^2	Y_2^2	Y_3^2	...	Y_{2n}^2	Y_{2n+1}^2
D9	Y_0^1	Y_1^1	Y_2^1	Y_3^1	...	Y_{2n}^1	Y_{2n+1}^1
D8	Y_0^0	Y_1^0	Y_2^0	Y_3^0	...	Y_{2n}^0	Y_{2n+1}^0
D7	U_0^7	V_0^7	U_2^7	V_2^7	...	U_{2n}^7	V_{2n+1}^7
D6	U_0^6	V_0^6	U_2^6	V_2^6	...	U_{2n}^6	V_{2n+1}^6
D5	U_0^5	V_0^5	U_2^5	V_2^5	...	U_{2n}^5	V_{2n+1}^5
D4	U_0^4	V_0^4	U_2^4	V_2^4	...	U_{2n}^4	V_{2n+1}^4
D3	U_0^3	V_0^3	U_2^3	V_2^3	...	U_{2n}^3	V_{2n+1}^3
D2	U_0^2	V_0^2	U_2^2	V_2^2	...	U_{2n}^2	V_{2n+1}^2
D1	U_0^1	V_0^1	U_2^1	V_2^1	...	U_{2n}^1	V_{2n+1}^1
D0	U_0^0	V_0^0	U_2^0	V_2^0	...	U_{2n}^0	V_{2n+1}^0

bit 8 Reserved
The default value for this bit is 0b.

bit 7 YRC Software Reset
This bit performs a software reset of YRC (YUV to RGB Converter) logic. It has no effect on the YRC registers.
Writing a 0b to this bit has no hardware effect.
Writing a 1b to this bit initiates a software reset of the YRC logic. This bit must be set back to 0b before the YRC is ready for use again.

bits 6-5 YRC UV Fix Select bits [1:0]
These bits control the UV input to YRC (YUV to RGB Converter) by allowing a the U data, V data, or both, to be “fixed” to a specified value as set in the YRC UV Data Fix register, REG[300Eh]. These bits have an effect on the UV data even when the YRC is bypassed, REG[3000h] bit 15 = 1b.

Table 10-98: YRC UV Fix Selection

REG[3000h] bits 6-5	UV Input to the YRC
00b	Original U data, original V data
01b	U data = REG[300Eh] bits 15-8, original V data
10b	Original U data, V data = REG[300Eh] bits 7-0
11b	U data = REG[300Eh] bits 15-8, V data = REG[300Eh] bits 7-0

bit 4 YRC Input Data Type Select

This bit specifies the input data type to the YRC (YUV to RGB Converter).

Table 10-99: YRC Input Data Type Selection

REG[3000h] bit 4	Input Data Type	Data Range
0b	YUV	0 =< Y =< 255 0 =< U =< 255 0 =< V =< 255
1b	YCbCr	16 =< Y =< 235 16 =< U =< 240 16 =< V =< 240

For REG[3000h] bit 4 = 0b.

Y’ = Y
U’ = U - 128
V’ = V - 128

For REG[3000h] bit 4 = 1b.

Y’ = (Y - 16) ÷ (235 - 16) x 255
U’ = (Cb - 128) ÷ (240 - 16) x 255
V’ = (Cr - 128) ÷ (240 - 16) x 255

These conversions are done to ensure the input to the YRC meets the following restrictions.

$0 \leq Y' \leq 255$
$-128 \leq U' \leq 127$
$-128 \leq V' \leq 127$

If YUV data exceeds the YRC restrictions, the YRC will not function properly and will set all values to 255.

bits 2-0 YRC Transfer Mode bits [2:0]

These bits specify the transfer mode used by the YRC (YUV to RGB Converter). Recommended settings are provided for various specifications.

Table 10-100: YRC Transfer Mode Selection

REG[3000h] bits 2-0	YUV to RGB Specification
000b	Reserved
001b	Recommended for ITU-R BT.709
010b	Reserved
011b	Reserved
100b	Recommended for ITU-R BT.470-6 System M
101b (Default)	Recommended for ITU-R BT.470-6 System B, G (Recommended for ITU-R BT.601-5)
110b	SMPTE 170M
111b	SMPTE 240M(1987)

REG[3002h] YRC Write Start Address 0 Register 0
Default = 0000h Read/Write

YRC Write Start Address 0 bits 15-8							
15	14	13	12	11	10	9	8
YRC Write Start Address 0 bits 7-0							
7	6	5	4	3	2	1	0

REG[3004h] YRC Write Start Address 0 Register 1
Default = 0000h Read/Write

n/a						YRC Write Start Address 0 bits 25-24	
15	14	13	12	11	10	9	8
YRC Write Start Address 0 bits 23-16							
7	6	5	4	3	2	1	0

REG[3004h] bits 9-0
REG[3002h] bits 15-0 YRC Write Start Address 0 bits [25:0]

These bits are only used for writes from the YRC (YUV to RGB Converter) to the external SDRAM memory. For single buffer write mode (REG[3000h] bit 12 = 0b), these bits determine the start address where the YRC (YUV to RGB Converter) writes data. For double buffer write mode (REG[3000h] bit 12 = 1b), these bits determine start address of the first buffer where the YRC writes data. The YRC writes data to the memory in 32-bit blocks, therefore bits 1-0 of this register must be set to 00b.

REG[3006h] YRC Write Start Address 1 Register 0
Default = 0000h Read/Write

YRC Write Start Address 1 bits 15-8							
15	14	13	12	11	10	9	8
YRC Write Start Address 1 bits 7-0							
7	6	5	4	3	2	1	0

REG[3008h] YRC Write Start Address 1 Register 1
Default = 0000h Read/Write

n/a						YRC Write Start Address 1 bits 25-24	
15	14	13	12	11	10	9	8
YRC Write Start Address 1 bits 23-16							
7	6	5	4	3	2	1	0

REG[3008h] bits 9-0
REG[3006h] bits 15-0 YRC Write Start Address 1 bits [25:0]

These bits are only used for writes from the YRC (YUV to RGB Converter) to the external SDRAM memory. For single buffer write mode (REG[3000h] bit 12 = 0b), these bits are not used. For double buffer write mode (REG[3000h] bit 12 = 1b), these bits determine start address of the second buffer where the YRC writes data. The YRC writes data to the memory in 32-bit blocks, therefore bits 1-0 of this register must be set to 00b.

REG[300Ah] through REG[300Ch] Reserved

These registers are Reserved and should not be written.

REG[300Eh] YRC UV Data Fix Register
Default = 0000h Read/Write

YRC U Data Fix bits 7-0							
15	14	13	12	11	10	9	8
YRC V Data Fix bits 7-0							
7	6	5	4	3	2	1	0

bits 15-8 YRC U Data Fix bits [7:0]
These bits only have an effect when the YRC UV Fix Select bits are set to 01b or 11b (REG[3000h] bits 6-5 = 01b or 11b). The U data input to YRC (YUV to RGB Converter) is fixed to the value of these bits.

bits 7-0 YRC V Data Fix bits [7:0]
These bits only have an effect when the YRC UV Fix Select bits are set to 10b or 11b (REG[3000h] bits 6-5 = 10b or 11b). The V data input to YRC (YUV to RGB Converter) is fixed to the value of these bits.

REG[3010h] YRC Rectangular Pixel Width Register
Default = 0000h Read/Write

n/a					YRC Rectangular Pixel Width bits 10-8		
15	14	13	12	11	10	9	8
YRC Rectangular Pixel Width bits 7-0							
7	6	5	4	3	2	1	0

bits 10-0 YRC Rectangular Pixel Width bits [10:0]
These bits are only used for writes from the YRC (YUV to RGB Converter) to the external SDRAM memory. These bits specify the horizontal pixel width of the data being written when the YRC (YUV to RGB Converter) is configured for rectangular write mode, REG[3000h] bit 14 = 1b.
For a color depth of 16 bpp, the rectangular pixel width must be an even number of pixels as only bits 10-1 are used.

REG[3012h] YRC Rectangular Line Address Offset Register
Default = 0000h Read/Write

n/a				YRC Rectangular Line Address Offset bits 11-8			
15	14	13	12	11	10	9	8
YRC Rectangular Line Address Offset bits 7-0							
7	6	5	4	3	2	1	0

bits 11-0 YRC Rectangular Line Address Offset bits [11:0]
These bits are only used for writes from the YRC (YUV to RGB Converter) to the external SDRAM memory. These bits specify the number of pixels from the beginning of the current display line to the beginning of the next line when YRC (YUV to RGB Converter) is configured for rectangular write mode, REG[3000h] bit 14 = 1b.
For a color depth of 16 bpp, the rectangular line address offset must be an even number of pixels (only bits 11-1 are used).
When the YRC is disabled, the rectangular line address offset may be an even or odd number of pixels as all of bits 11-0 are used.

REG[3014h] YRC Memory Configuration Register
Default = 0000h Read Only

Reserved	n/a			Reserved	n/a		
15	14	13	12	11	10	9	8
n/a			Reserved	n/a	YRC Frame Buffer Write Mode Status bits 1-0 (RO)		YRC RAM Interface Data Write Status (RO)
7	6	5	4	3	2	1	0

bit 15 Reserved
The default value for this bit is 0b.

bit 11 Reserved
The default value for this bit is 0b.

bit 4 Reserved
The default value for this bit is 0b.

bits 2-1 YRC Frame Buffer Write Mode Status bits [1:0] (Read Only)
These bits indicate which buffer in memory the YRC is writing data to.

Table 10-101: YRC Frame Buffer Write Mode Status

REG[3014h] bits 2-1	Data Write Mode
00b	First Buffer (REG[3002h] ~ REG[3004h])
01b	Second Buffer (REG[3006h] ~ REG[3008h])
10b - 11b	Reserved

bit 0 YRC Data Write Status (Read Only)
This bit indicates the status of YRC data writes to the SDRAM memory.
When this bit = 0b, the YRC is not currently writing data to the SDRAM memory.
When this bit = 1b, the YRC is currently writing data to the SDRAM memory.

10.4.16 PWM Registers

Note

The pins used by the PWM interface are multiplexed with GPIO function pins. Therefore, before enabling the PWM interface, the appropriate GPIO pins must be configured for use by the PWM interface. For a summary of GPIO pin usage, see Section 5.6, “GPIO Pin Mapping” on page 50.

REG[3400h] PWM Control Register
Default = 0000h Read/Write

Reserved				Blue Enable	Green Enable	Red Enable	White Enable
15	14	13	12	11	10	9	8
n/a	Reserved	AUDIN Active State Select	AUDIN Control Enable	n/a	Output Polarity	PWM RGB Output Enable	PWM White Output Enable
7	6	5	4	3	2	1	0

Note

If REG[3400h] bits 11-8 equal 0000b (all PWM outputs are disabled), the clock to the entire PWM circuit is dynamically disabled in order to save power and minimize current drain.

bits 15-12 Reserved
The default value for these bits is 0000b.

bit 11 Blue Enable
This bit controls the Blue LED PWM output. For PWM pin mapping, see Section 5.10, “PWM Interface Pin Mapping” on page 52.
When this bit = 0b, the PWMB output is disabled (becomes logic 0 before the polarity inversion circuit specified by REG[3400h] bit 2).
When this bit = 1b, the PWMB output is enabled.

Note

If all of the Red Enable, Green Enable, and Blue Enable bits are disabled, then the 128 clock reference point described in Section 19.1, “PWM Circuit Overview” on page 429 is reset to zero.

bit 10 Green Enable
This bit controls the Green LED PWM output. For PWM pin mapping, see Section 5.10, “PWM Interface Pin Mapping” on page 52.
When this bit = 0b, the PWMG output is disabled (becomes logic 0 before the polarity inversion circuit specified by REG[3400h] bit 2).
When this bit = 1b, the PWMG output is enabled.

Note

If all of the Red Enable, Green Enable, and Blue Enable bits are disabled, then the 128 clock reference point described in Section 19.1, “PWM Circuit Overview” on page 429 is reset to zero.

bit 9 Red Enable
This bit controls the Red LED PWM output. For PWM pin mapping, see Section 5.10, “PWM Interface Pin Mapping” on page 52.
When this bit = 0b, the PWMR output is disabled (becomes logic 0 before the polarity inversion circuit specified by REG[3400h] bit 2).
When this bit = 1b, the PWMR output is enabled.

Note

If all of the Red Enable, Green Enable, and Blue Enable bits are disabled, then the 128 clock reference point described in Section 19.1, “PWM Circuit Overview” on page 429 is reset to zero.

bit 8 White Enable
This bit controls the White LED PWM output. For PWM pin mapping, see Section 5.10, “PWM Interface Pin Mapping” on page 52.
When this bit = 0b, the PWMW output is disabled (becomes logic 0 before the polarity inversion circuit specified by REG[3400h] bit 2).
When this bit = 1b, the PWMW output is enabled.

bit 6 Reserved
The default value for this bit is 0b.

bit 5 AUDIN Active State Select
When AUDIN control of the PWM outputs is enabled (REG[3400h] bit 4 = 1b), this bit selects the active state of the digital audio input, AUDIN. For PWM pin mapping, see Section 5.10, “PWM Interface Pin Mapping” on page 52.
When this bit = 0b, a high on the digital audio input enables the color PWM outputs.
When this bit = 1b, a low on the digital audio input enables the color PWM outputs.

Note

This bit should be used in conjunction with REG[0C0Eh] bits 7-6 to set the function of GPIOD3.

bit 4 AUDIN Control Enable
This bit enables/disables control of the three color PWM outputs (PWMB, PWMG, PWMR) by the digital audio input, AUDIN. For PWM pin mapping, see Section 5.10, “PWM Interface Pin Mapping” on page 52.
When this bit = 0b, the digital audio input does not control the PWM outputs.
When this bit = 1b, the digital audio input controls the PWM outputs.

Note

Before using the AUDIN input, the appropriate GPIO pin must be configured for use by the PWM interface.

bit 2 Output Polarity
This bit specifies the polarity of the output pin relative to the digital value output by the PWM circuit for all 4 LED output pins (three color and one white).
When this bit = 0b, the LED pin voltage is driven low when a logic 1 is driven from the PWM circuit and driven high when a logic 0 is driven from the PWM circuit.
When this bit = 1b, the LED output pin is driven high when the PWM circuit is driving a logic 1.

bit 1 PWM RGB Output Enable
This bit controls the three color LED PWM outputs (PWMB, PWMG, PWMR).
When this bit = 0b, the color LED PWM outputs are held high.
When this bit = 1b, the color LED PWM outputs are enabled.

Note

If both bit 0 and bit 1 are zero (i.e. both PWM outputs are held high), then the clock to the entire PWM circuit is turned off (to save power and minimize current drain) once all PWM Enable bits are disabled (see REG[3400h] bits 11-8).

bit 0 PWM White Output Enable
This bit controls the white LED PWM output (PWMW).
When this bit = 0b, the white LED PWM output is held high.
When this bit = 1b, the white LED PWM output is enabled.

Note

If both bit 0 and bit 1 are zero (i.e. both PWM outputs are held high), then the clock to the entire PWM circuit is turned off (to save power and minimize current drain) once all PWM Enable bits are disabled (see REG[3400h] bits 11-8).

REG[3402h] PWM Clock Divide Register
Default = 0000h Read/Write

n/a							
15	14	13	12	11	10	9	8
n/a				PWM Clock Divide Select bits 3-0			
7	6	5	4	3	2	1	0

bits 3-0 PWM Clock Divide Select bits [3:0]
These bits select the clock divide for the PWM clock used by the PWMR, PWMG and PWMB circuits. It has no effect on PWMW. The clock source for the PWM clock is the internal clock PWMSRCCLK. PWMSRCCLK should be configured to provide an approximately 16KHz clock which allows for an LED pulse rate ranging from 0.5Hz to 8Hz. For further details on PWMSRCLK, see Section 9, “Clocks” on page 124.

Table 10-102: PWM Clock Divide Selection

REG[3402h] bits 3-0	PWM Clock Divide Ratio
0000b	1:1
0001b	2:1
0010b	4:1
0011b	6:1
0100b	8:1
0101b	10:1
0110b	12:1
0111b	14:1
1000b	16:1
1001b - 1111b	Reserved (PWM Clock is stopped)

REG[3404h] Red On/Off Control Register
Default = 0000h Read/Write

n/a	Red Off bits 6-0						
15	14	13	12	11	10	9	8
n/a	Red On bits 6-0						
7	6	5	4	3	2	1	0

bits 14-8 Red Off bits [6:0]
These bits specify the point at which the red LED turns off relative to the start of the 128 clock pulse cycle. This value must be greater than that of the “Red On” value specified in REG[3404h] bits 6-0 or unpredictable results may occur. For further information on using PWM, see Section 19, “Pulse Width Modulation (PWM)” on page 429.

REG[3404h] bits 14-8 = Off Time - 1

Note

If a value of 7Fh is entered, the LED is on for the entire duration of the red duty cycle, REG[340Ch] bits 3-0.

bits 6-0 Red On bits [6:0]
These bits specify the point at which the red LED turns on relative to the start of the 128 clock pulse cycle. A value of 0 means the LED starts the turn on sequence immediately at the start of the 128 clock cycle. For further information on using PWM, see Section 19, “Pulse Width Modulation (PWM)” on page 429.

REG[3406h] Green On/Off Control Register
Default = 0000h Read/Write

n/a	Green Off bits 6-0						
15	14	13	12	11	10	9	8
n/a	Green On bits 6-0						
7	6	5	4	3	2	1	0

bits 14-8 Green Off bits [6:0]
These bits specify the point at which the green LED turns off relative to the start of the 128 clock pulse cycle. This value must be greater than that of the “Green On” value specified in REG[3406h] bits 6-0 or unpredictable results may occur. For further information on using PWM, see Section 19, “Pulse Width Modulation (PWM)” on page 429.
REG[3406h] bits 14-8 = Off Time - 1

Note
If a value of 7Fh is entered, the LED is on for the entire duration of the green duty cycle, REG[340Ch] bits 7-4.

bits 6-0 Green On bits [6:0]
These bits specify the point at which the green LED turns on relative to the start of the 128 clock pulse cycle. A value of 0 means the LED starts the turn on sequence immediately at the start of the 128 clock cycle. For further information on using PWM, see Section 19, “Pulse Width Modulation (PWM)” on page 429.

REG[3408h] Blue On/Off Control Register
Default = 0000h Read/Write

n/a	Blue Off bits 6-0						
15	14	13	12	11	10	9	8
n/a	Blue On bits 6-0						
7	6	5	4	3	2	1	0

bits 14-8 Blue Off bits [6:0]
These bits specify the point at which the blue LED turns off relative to the start of the 128 clock pulse cycle. This value must be greater than that of the “Blue On” value specified in REG[3408h] bits 6-0 or unpredictable results may occur. For further information on using PWM, see Section 19, “Pulse Width Modulation (PWM)” on page 429.
REG[3408h] bits 14-8 = Off Time - 1

Note
If a value of 7Fh is entered, the LED is on for the entire duration of the blue duty cycle, REG[340Ch] bits 11-8.

bits 6-0 Blue On bits [6:0]
These bits specify the point at which the blue LED turns on relative to the start of the 128 clock pulse cycle. A value of 0 means the LED starts the turn on sequence immediately at the start of the 128 clock cycle. For further information on using PWM, see Section 19, “Pulse Width Modulation (PWM)” on page 429.

REG[340Ah] PWM Slope Register Default = 0000h							Read/Write
n/a	LED Pulse Counter (M) bits 2-0			Blue Slope bits 3-0			
15	14	13	12	11	10	9	8
Green Slope bits 3-0				Red Slope bits 3-0			
7	6	5	4	3	2	1	0

Note

Using a slope that is not divisible by the color’s maximum duty cycle before the down slope will result in an asymmetrical signal.

bits 14-12 LED Pulse Counter (M) bits [2:0]
These bits determine the M value used for the slope calculation. At every M+1 clocks of the 128 clock wide LED pulse the duty cycle increases by a value (1/16 x N), where N is determined by the corresponding Red (bits 3-0), Green (bits 7-4), and Blue (bits 11-8) Slope bits. These bits have no effect when the Slope bits for a particular color are set to 0.

REG[340Ah] bits 14-12 = M value for slope calculation - 1

bits 11-8 Blue Slope bits [3:0]
These bits specify the rate of change at which the duty cycle changes as the blue LED goes from completely off to the maximum duty cycle specified in REG[340Ch] bits 11-8. At every output pulse (M+1) of 128 pulse counter (M is specified in REG[340Ah] bits 14-12), the duty cycle increases by (1/16 x N) where N is the decimal value represented by these bits. If these bits are set to a value of 0h, the duty cycle immediately changes from completely off, to the maximum duty cycle as specified by the Blue Duty Cycle bits, REG[340Ch] bits 11-8.

bits 7-4 Green Slope bits [3:0]
These bits specify the rate of change at which the duty cycle changes as the green LED goes from completely off to the maximum duty cycle specified in REG[340Ch] bits 7-4. At every output pulse (M+1) of 128 pulse counter (M is specified in REG[340Ah] bits 14-12), the duty cycle increases by (1/16 x N) where N is the decimal value represented by these bits. If these bits are set to a value of 0h, the duty cycle immediately changes from completely off, to the maximum duty cycle as specified by the Green Duty Cycle bits, REG[340Ch] bits 7-4.

bits 3-0 Red Slope bits [3:0]
These bits specify the rate of change at which the duty cycle changes as the red LED goes from completely off to the maximum duty cycle specified in REG[340Ch] bits 3-0. At every output pulse (M+1) of 128 pulse counter (M is specified in REG[340Ah] bits 14-12), the duty cycle increases by (1/16 x N) where N is the decimal value represented by these bits. If these bits are set to a value of 0h, the duty cycle immediately changes from completely off, to the maximum duty cycle as specified by the Red Duty Cycle bits, REG[340Ch] bits 3-0.

REG[340Ch] PWM Duty Cycle Register Default = 0000h							Read/Write
n/a				Blue Duty Cycle bits 3-0			
15	14	13	12	11	10	9	8
Green Duty Cycle bits 3-0				Red Duty Cycle bits 3-0			
7	6	5	4	3	2	1	0

Note

Using a slope that is not divisible by the color’s maximum duty cycle before the down slope will result in an asymmetrical signal.

bits 11-8 Blue Duty Cycle bits [3:0]
These bits specify the “full-on” duty cycle which determines the maximum brightness that the LED reaches at the peak of the pulse. A value of Fh indicates full brightness (i.e. continuously on). A value of 0 means the LED is on for 1/16th of the time.

Note

When the blue slope (REG[340Ah] bits 11-8) is non-zero, the blue duty cycle must not be set to 1111b (Fh).

bits 7-4 Green Duty Cycle bits [3:0]
These bits specify the “full-on” duty cycle which determines the maximum brightness that the LED reaches at the peak of the pulse. A value of Fh indicates full brightness (i.e. continuously on). A value of 0 means the LED is on for 1/16th of the time.

Note

When the green slope (REG[340Ah] bits 7-4) is non-zero, the green duty cycle must not be set to 1111b (Fh).

bits 3-0 Red Duty Cycle bits [3:0]
These bits specify the “full-on” duty cycle which determines the maximum brightness that the LED reaches at the peak of the pulse. A value of Fh indicates full brightness (i.e. continuously on). A value of 0 means the LED is on for 1/16th of the time.

Note

When the red slope (REG[340Ah] bits 3-0) is non-zero, the red duty cycle must not be set to 1111b (Fh).

REG[340Eh] White LED Control Register
Default = 0000h Read/Write

n/a		White LED Duty Cycle bits 5-0					
15	14	13	12	11	10	9	8
n/a		White LED Period bits 5-0					
7	6	5	4	3	2	1	0

bits 13-8 White LED Duty Cycle bits [5:0]

These bits specify the duty cycle, or “on” time, for the White LED. A value of 00h represents the dimmest LED brightness (the shortest time which the LED can be “on” for each period as specified in REG[340Eh] bits 5-0. When the white LED duty cycle is set to 3Fh, the LED is continuously “on”.

REG[340Eh] bits 13-8 = white LED duty cycle - 1

Note

The output of the white LED can be forced to an “off” state using the PWM White Output Enable bit (REG[3400h] bit 0) or the White Enable bit (REG[3400h] bit 8).

bits 5-0 White LED Period bits [5:0]

These bits specify the period of the white LED PWM output. For a PWMSRCCLK of approximately 16kHz, the following formula provides a frequency range of between 64Hz to 1Hz for the white LED.

PWM period = 256 x (1 ÷ PWMSRCCLK) x ((REG[340Eh] bits 5-0) + 1)

REG[3410h] through REG[3412h] are Reserved

These registers are Reserved and should not be written.

10.4.17 I2C Registers

Note

The pins used by the I2C interface are multiplexed with GPIO function pins. Therefore, before using the I2C interface, the appropriate GPIO pins must be configured for use by the I2C interface. For a summary of GPIO pin usage, see Section 5.6, “GPIO Pin Mapping” on page 50.

REG[3800h] I2C Control Register
(Default = 0065h) Read/Write

n/a							
15	14	13	12	11	10	9	8
n/a	Reserved	I2C Restart Enable	n/a	Reserved	I2C Speed Select bits 1-0		I2C Master Enable
7	6	5	4	3	2	1	0

Note

The I2C must be disabled (REG[386Ch] bit 1 = 0b) before writing to these bits. Writing to these bits when the I2C is enabled has no effect.

bit 6 Reserved
This bit must be set to 1b.

bit 5 I2C Restart Enable
This bit determines whether the master can send restart conditions. Disabling restart conditions is required for some older slaves that do not support handling restart conditions. However, restart conditions are used for several I2C operations so when restart conditions are disabled, the master cannot perform the following functions.

- send multiple bytes per transfer (split)
- change direction within a transfer (split)
- send a start byte

Split operations are broken down into multiple I2C transfers with a stop and start condition in between. The other operations are not performed at all and result in a Transmit Abort interrupt (see REG[382Ch] bit 6).
When this bit = 0b, the master cannot send restart conditions (disabled).
When this bit = 1b, the master can send restart conditions (enabled). (default)

bit 3 Reserved
The default value for this bit is 0b.

bits 2-1 I2C Speed Select bits [1:0]
These bits select the operating speed for the I2C interface The operating speed can only be changed when the I2C interface is disabled, REG[386Ch] bit 0 = 0b.

Table 10-103 : I2C Speed Selection

REG[3800h] bits 2-1	I2C Speed
00b	Reserved
01b	Standard Mode (100kbs)
10b	Fast Mode (400kbs)
11b	Reserved

bit 0 I2C Master Enable
This bit controls the I2C master and can only be changed when the I2C interface is disabled, REG[386Ch] bit 0 = 0b.
When this bit = 0b, the I2C master is disabled.
When this bit = 1b, the I2C master is enabled. (default)

REG[3804h] I2C Target Address Register
(Default = 0055h) Read/Write

n/a						Reserved	
15	14	13	12	11	10	9	8
Reserved	I2C Target Slave Address bits 6-0						
7	6	5	4	3	2	1	0

Note

The I2C must be disabled (REG[386Ch] bit 1 = 0b) before writing to these bits. Writing to these bits when the I2C is enabled has no effect.

bits 9-7 Reserved
The default value for these bits is 000b.

bits 6-0 I2C Target Slave Address bits [6:0]
These bits are the 7-bit target slave address used for master transactions.

REG[3808h] is Reserved

This register is Reserved and should not be written.

REG[3810h] I2C Receive/Transmit Data Buffer and Command Register
(Default = 0000h) Read/Write

n/a							I2C Command (WO)
15	14	13	12	11	10	9	8
I2C Data bits 7-0							
7	6	5	4	3	2	1	0

bit 8 I2C Command (Write Only)
This bit determines whether a read or a write is performed by the I2C interface.
Writing a 0b to this bit selects a write operation.
Writing a 1b to this bit selects a read operation.

bits 7-0 I2C Data bits [7:0]
These bits contain the data that will be transmitted to or received from the I2C bus. For data writes (REG[3810h] bit 8 = 0b), the data to be send must be placed in these bits prior to the transfer. For data reads (REG[3810h] bit 8 = 1b), the data received from the I2C interface can be read from these bits.

REG[3814h] I2C Standard Speed I2C Clock SCL High Count Register
(Default = 0064h) Read/Write

I2C Standard Speed SCL High Count bits 15-8							
15	14	13	12	11	10	9	8
I2C Standard Speed SCL High Count bits 7-0							
7	6	5	4	3	2	1	0

bits 15-0 I2C Standard Speed SCL High Count bits [15:0]
To ensure correct I2C interface IO timing, these bits must be set before any I2C bus transaction can take place. These bits set the SCL clock high-period count for standard speed. The I2C interface must be disabled (REG[386Ch] bit 0 = 0b) before writing to these bits, or writes will have no effect. The minimum value for these bits is 6. If a value less than 6 is written, the value will be ignored and the register will be set to 6.

SCL High Count time = T_{I2C} x (REG[3814h] bits 15-0 + 8)

The following table provides some example values.

Table 10-104 : I2C Standard Speed High Count Example Values

I2C Clock Frequency (MHz)	T_{I2C} (ns)	REG[0430h]	I2C Divide Ratio	REG[3814h]	SCL High Time (µs)
1	1000	0031h	50	0006h	14.00
2	500	0018h	25	0006h	7.00
5	200	0009h	10	000Ch	4.00
10	100	0004h	5	0020h	4.00
12.5	80	0003h	4	002Ah	4.00
16.6	60	0002h	3	003Bh	4.02
25	40	0001h	2	005Ch	4.00
50	20	0000h	1	00C0h	4.00

REG[3818h] I2C Standard Speed I2C Clock SCL Low Count Register
(Default = 0076h) Read/Write

I2C Standard Speed SCL Low Count bits 15-8							
15	14	13	12	11	10	9	8

I2C Standard Speed SCL Low Count bits 7-0							
7	6	5	4	3	2	1	0

bits 15-0 I2C Standard Speed SCL Low Count bits [15:0]
To ensure correct I2C interface IO timing, these bits must be set before any I2C bus transaction can take place. These bits set the SCL clock low-period count for standard speed. The I2C interface must be disabled (REG[386Ch] bit 0 = 0b) before writing to these bits, or writes will have no effect. The minimum value for these bits is 8. If a value less than 8 is written, the value will be ignored and the register will be set to 8.

SCL Low Count time = T_{I2C} x (REG[3818h] bits 15-0 + 1)

The following table provides some example values.

Table 10-105 : I2C Standard Speed Low Count Example Values

I2C Clock Frequency (MHz)	T_{I2C} (ns)	REG[0430h]	I2C Divide Ratio	REG[3818h]	SCL Low Time (µs)
1	1000	0031h	50	0008h	9.00
2	500	0018h	25	0009h	5.00
5	200	0009h	10	0017h	4.80
10	100	0004h	5	002Eh	4.70
12.5	80	0003h	4	003Ah	4.72
16.6	60	0002h	3	004Eh	4.74
25	40	0001h	2	0074h	4.72
50	20	0000h	1	00EAh	4.70

I

REG[381Ch] I2C Fast Speed I2C Clock SCL High Count Register
(Default = 000Fh) Read/Write

I2C Fast Speed SCL High Count bits 15-8							
15	14	13	12	11	10	9	8
I2C Fast Speed SCL High Count bits 7-0							
7	6	5	4	3	2	1	0

bits 15-0 I2C Fast Speed SCL High Count bits [15:0]
To ensure correct I2C interface IO timing, these bits must be set before any I2C bus transaction can take place. These bits set the SCL clock high-period count for fast speed. It is used in high-speed mode to send the Master Code and START BYTE or General CALL. The I2C interface must be disabled (REG[386Ch] bit 0 = 0b) before writing to these bits, or writes will have no effect. The minimum value for these bits is 6. If a value less than 6 is written, the value will be ignored and the register will be set to 6.

SCL High Count time = T_{I2C} x (REG[381Ch] bits 15-0 + 8)

The table below provides some example values.

Table 10-106 : I2C Fast Speed High Count Example Values

I2C Clock Frequency (MHz)	T_{I2C} (ns)	REG[0430h]	I2C Divide Ratio	REG[381Ch]	SCL High Time (µs)
1	1000	0031h	50	0006h	14.00
2	500	0018h	25	0006h	7.00
5	200	0009h	10	0006h	2.80
10	100	0004h	5	0006h	1.40
12.5	80	0003h	4	0006h	1.12
16.6	60	0002h	3	0008h	0.96
25	40	0001h	2	0010h	0.96
50	20	0000h	1	0028h	0.96

REG[3820h] I2C Fast Speed I2C Clock SCL Low Count Register (Default = 0021h)							Read/Write
I2C Fast Speed SCL Low Count bits 15-8							
15	14	13	12	11	10	9	8
I2C Fast Speed SCL Low Count bits 7-0							
7	6	5	4	3	2	1	0

bits 15-0

I2C Fast Speed SCL Low Count bits [15:0]
To ensure correct I2C interface IO timing, these bits must be set before any I2C bus transaction can take place. These bits set the SCL clock low-period count for fast speed. It is used in high-speed mode to send the Master Code and START BYTE or General CALL. The I2C interface must be disabled (REG[386Ch] bit 0 = 0b) before writing to these bits, or writes will have no effect. The minimum value for these bits is 8. If a value less than 86 is written, the value will be ignored and the register will be set to 8.

SCL Low Count time = T_{I2C} x (REG[3820h] bits 15-0 + 1)

The table below provides some example values.

Table 10-107 : I2C Fast Speed Low Count Example Values

I2C Clock Frequency (MHz)	T_{I2C} (ns)	REG[0430h]	I2C Divide Ratio	REG[3820h]	SCL Low Time (µs)
1	1000	0031h	50	0008h	9.00
2	500	0018h	25	0008h	4.50
5	200	0009h	10	0008h	1.80
10	100	0004h	5	000Dh	1.40
12.5	80	0003h	4	000Dh	1.12
16.6	60	0002h	3	000Fh	0.96
25	40	0001h	2	0017h	0.96
50	20	0000h	1	002Fh	0.96

REG[3824h] through REG[3828h] are Reserved

These registers are Reserved and should not be written.

REG[382Ch] I2C Interrupt Status Register (Default = 0000h)							Read Only
n/a					I2C Start Interrupt Status	I2C Stop Interrupt Status	I2C Busy Interrupt Status
15	14	13	12	11	10	9	8
Reserved	I2C Transmit Abort Interrupt Status	Reserved	I2C Transmit FIFO Empty Interrupt Status	I2C Transmit FIFO Overflow Interrupt Status	I2C Receive FIFO Full Interrupt Status	I2C Receive FIFO Overflow Interrupt Status	I2C Receive FIFO Underflow Interrupt Status
7	6	5	4	3	2	1	0

bit 10 I2C Start Interrupt Status (Read Only)
This bit indicates the status of the I2C Start Interrupt which is triggered when a start condition occurs on the I2C interface. This bit is masked by the I2C Start Interrupt Enable bit and is only available when REG[3830h] bit 10 = 1b. A raw (unmasked) status bit is available in REG[3834h].
When this bit = 0b, an I2C Start Interrupt has not occurred.
When this bit = 1b, an I2C Start Interrupt has occurred.

To clear this status bit, read the I2C Start Interrupt Clear bit, REG[3864h] bit 0.

bit 9 I2C Stop Interrupt Status (Read Only)
This bit indicates the status of the I2C Stop Interrupt which is triggered when a stop condition occurs on the I2C interface. This bit is masked by the I2C Stop Interrupt Enable bit and is only available when REG[3830h] bit 9 = 1b. A raw (unmasked) status bit is available in REG[3834h].
When this bit = 0b, an I2C Stop Interrupt has not occurred.
When this bit = 1b, an I2C Stop Interrupt has occurred.

To clear this status bit, read the I2C Stop Interrupt Clear bit, REG[3860h] bit 0.

bit 8 I2C Busy Interrupt Status (Read Only)
This bit indicates the status of the I2C Busy Interrupt which is triggered when there is activity on the I2C interface (I2C interface is busy). This bit is masked by the I2C Busy Interrupt Enable bit and is only available when REG[3830h] bit 8 = 1b. A raw (unmasked) status bit is available in REG[3834h].
When this bit = 0b, an I2C Busy Interrupt has not occurred.
When this bit = 1b, an I2C Busy Interrupt has occurred.

To clear this status bit, read the I2C Busy Interrupt Clear bit, REG[385Ch] bit 0.

bit 7 Reserved
The default value for this bit is 0b.

bit 6 I2C Transmit Abort Interrupt Status (Read Only)
This bit indicates the status of the I2C Transmit Abort Interrupt which triggers when the I2C acting as a master is unable to complete a command that the Host has sent. This bit is masked by the I2C Transmit Abort Interrupt Enable bit and is only available when REG[3830h] bit 6 = 1b. A raw (unmasked) status bit is available in REG[3834h].
When this bit = 0b, an I2C Transmit Abort Interrupt has not occurred.
When this bit = 1b, an I2C Transmit Abort Interrupt has occurred.

To clear this status bit, read the I2C Transmit Abort Interrupt Clear bit, REG[3854h] bit 0.

The specific condition that caused the interrupt is indicated in the I2C Transmit Abort Source register (REG[3880h]) and identifies the following conditions.

- no slave acknowledges after the address is sent.
- the addressed slave does not acknowledge a byte of data.
- arbitration is lost.
- attempting to send a master command when configured only as a slave.
- IC_RESTART_EN bit in the IC_CON register is set to 0 (restart condition disabled), and the processor attempts to issue an I2C function that is impossible to perform without using restart conditions.
- high-speed master code is acknowledge.
- start byte is acknowledged.
- general call address is not acknowledged.
- when a read request interrupt occurs and the processor has previously placed data in the TX buffer that has not been transmitted yet. This data could have been intended to service a multi-byte RD_REQ that ended up having fewer numbers of bytes requested. Or, if IC_RESTART_EN is disabled and the I2C loses control of the bus between transfers and is then accessed as a slave-transmitter.
- if a read command is issued after a general call command has been issued. Disabling the I2C reverts it back to normal operation.
- if the processor attempts to issue read command before a RD_REQ is serviced. Anytime this bit is set, the contents of the transmit and receive buffers are flushed.

bit 5 Reserved
The default value for this bit is 0b.

bit 4

I2C Transmit FIFO Empty Interrupt Status (Read Only)
This bit indicates the status of the I2C Transmit FIFO Empty Interrupt which triggers when the Transmit FIFO level is at or below the level specified by the I2C Transmit FIFO Threshold Level bits (REG[383Ch] bits 2-0). This bit is masked by the I2C Transmit FIFO Empty Interrupt Enable bit and is only available when REG[3830h] bit 4 = 1b. A raw (unmasked) status bit is available in REG[3834h].
When this bit = 0b, an I2C Transmit FIFO Empty Interrupt has not occurred.
When this bit = 1b, an I2C Transmit FIFO Empty Interrupt has occurred.

This status bit is cleared automatically when the Transmit FIFO level goes above the Transmit FIFO Threshold Level, REG[3874h] bits 3-0.
transmit FIFO level > REG[3874h] bits 3-0

bit 3

I2C Transmit FIFO Overflow Interrupt Status (Read Only)
This bit indicates the status of the I2C Transmit FIFO Overflow Interrupt which triggers when the Transmit FIFO is completely filled and the Host attempts to issue another I2C command by writing to REG[3810h]. This bit is masked by the I2C Transmit FIFO Overflow Interrupt Enable bit and is only available when REG[3830h] bit 3 = 1b. A raw (unmasked) status bit is available in REG[3834h].
When this bit = 0b, an I2C Transmit FIFO Overflow Interrupt has not occurred.
When this bit = 1b, an I2C Transmit FIFO Overflow Interrupt has occurred.

To clear this status bit, read the I2C Transmit FIFO Overflow Interrupt Clear bit, REG[384Ch] bit 0.

bit 2

I2C Receive FIFO Full Interrupt Status (Read Only)
This bit indicates the status of the I2C Receive FIFO Full Interrupt which triggers when the Receive FIFO level reaches or exceeds the level specified by the I2C Receive FIFO Threshold Level bits (REG[3838h] bits 2-0). This bit is masked by the I2C Receive FIFO Full Interrupt Enable bit and is only available when REG[3830h] bit 2 = 1b. A raw (unmasked) status bit is available in REG[3834h].
When this bit = 0b, an I2C Receive FIFO Full Interrupt has not occurred.
When this bit = 1b, an I2C Receive FIFO Full Interrupt has occurred.

This status bit is cleared automatically when the Receive FIFO level goes below the Receive FIFO Threshold Level, REG[3878h] bits 3-0.
receive FIFO level > REG[3878h] bits 3-0

bit 1

I2C Receive FIFO Overflow Interrupt Status (Read Only)
This bit indicates the status of the I2C Receive FIFO Overflow Interrupt which is triggered when the Receive FIFO is completely full and more data arrives. When this happens, the data is lost. This bit is masked by the I2C Receive FIFO Overflow Interrupt Enable bit and is only available when REG[3830h] bit 1 = 1b. A raw (unmasked) status bit is available in REG[3834h].
When this bit = 0b, an I2C Receive FIFO Overflow Interrupt has not occurred.
When this bit = 1b, an I2C Receive FIFO Overflow Interrupt has occurred.

To clear this status bit, read the I2C Receive FIFO Overflow Interrupt Clear bit, REG[3848h] bit 0.

bit 0 I2C Receive FIFO Underflow Interrupt Status (Read Only)
This bit indicates the status of the I2C Receive FIFO Underflow Interrupt which is triggered when the Host attempts to read an empty Receive FIFO by reading from REG[3810h]. This bit is masked by the I2C Receive FIFO Underflow Interrupt Enable bit and is only available when REG[3830h] bit 0 = 1b. A raw (unmasked) status bit is available in REG[3834h].
When this bit = 0b, an I2C Receive FIFO Underflow Interrupt has not occurred.
When this bit = 1b, an I2C Receive FIFO Underflow Interrupt has occurred.

To clear this status bit, read the I2C Receive FIFO Underflow Interrupt Clear bit, REG[3844h] bit 0.

REG[3830h] I2C Interrupt Enable Register
(Default = 08FFh) Read/Write

n/a				Reserved	I2C Start Interrupt Enable	I2C Stop Interrupt Enable	I2C Busy Interrupt Enable
15	14	13	12	11	10	9	8
Reserved	I2C Transmit Abort Interrupt Enable	Reserved	I2C Transmit FIFO Empty Interrupt Enable	I2C Transmit FIFO Overflow Interrupt Enable	I2C Receive FIFO Full Interrupt Enable	I2C Receive FIFO Overflow Interrupt Enable	I2C Receive FIFO Underflow Interrupt Enable
7	6	5	4	3	2	1	0

bit 11 Reserved
The default value for this bit is 1b.

bit 10 I2C Start Interrupt Enable
This bit controls the I2C Start Interrupt and masks the corresponding status bit in REG[382Ch]. A raw (unmasked) status bit is available in REG[3834h].
When this bit = 0b, the I2C Start Interrupt is disabled.
When this bit = 1b, the I2C Start Interrupt is enabled.

bit 9 I2C Stop Interrupt Enable
This bit controls the I2C Stop Interrupt and masks the corresponding status bit in REG[382Ch]. A raw (unmasked) status bit is available in REG[3834h].
When this bit = 0b, the I2C Stop Interrupt is disabled.
When this bit = 1b, the I2C Stop Interrupt is enabled.

bit 8 I2C Busy Interrupt Enable
This bit controls the I2C Busy Interrupt and masks the corresponding status bit in REG[382Ch]. A raw (unmasked) status bit is available in REG[3834h].
When this bit = 0b, the I2C Busy Interrupt is disabled.
When this bit = 1b, the I2C Busy Interrupt is enabled.

bit 7 Reserved
The default value for this bit is 1b.

bit 6 I2C Transmit Abort Interrupt Enable
This bit controls the I2C Transmit Abort Interrupt and masks the corresponding status bit in REG[382Ch]. A raw (unmasked) status bit is available in REG[3834h].
When this bit = 0b, the I2C Transmit Abort Interrupt is disabled.
When this bit = 1b, the I2C Transmit Abort Interrupt is enabled.

bit 5 Reserved
The default value for this bit is 1b.

bit 4 I2C Transmit FIFO Empty Interrupt Enable
This bit controls the I2C Transmit FIFO Empty Interrupt and masks the corresponding status bit in REG[382Ch]. A raw (unmasked) status bit is available in REG[3834h].
When this bit = 0b, the I2C Transmit FIFO Empty Interrupt is disabled.
When this bit = 1b, the I2C Transmit FIFO Empty Interrupt is enabled.

bit 3 I2C Transmit FIFO Overflow Interrupt Enable
This bit controls the I2C Transmit FIFO Overflow Interrupt and masks the corresponding status bit in REG[382Ch]. A raw (unmasked) status bit is available in REG[3834h].
When this bit = 0b, the I2C Transmit FIFO Overflow Interrupt is disabled.
When this bit = 1b, the I2C Transmit FIFO Overflow Interrupt is enabled.

bit 2 I2C Receive FIFO Full Interrupt Enable
This bit controls the I2C Receive FIFO Full Interrupt and masks the corresponding status bit in REG[382Ch]. A raw (unmasked) status bit is available in REG[3834h].
When this bit = 0b, the I2C Receive FIFO Full Interrupt is disabled.
When this bit = 1b, the I2C Receive FIFO Full Interrupt is enabled.

bit 1 I2C Receive FIFO Overflow Interrupt Enable
This bit controls the I2C Receive FIFO Overflow Interrupt and masks the corresponding status bit in REG[382Ch]. A raw (unmasked) status bit is available in REG[3834h].
When this bit = 0b, the I2C Receive FIFO Overflow Interrupt is disabled.
When this bit = 1b, the I2C Receive FIFO Overflow Interrupt is enabled.

bit 0 I2C Receive FIFO Underflow Interrupt Enable
This bit controls the I2C Receive FIFO Underflow Interrupt and masks the corresponding status bit in REG[382Ch]. A raw (unmasked) status bit is available in REG[3834h].
When this bit = 0b, the I2C Receive FIFO Underflow Interrupt is disabled.
When this bit = 1b, the I2C Receive FIFO Underflow Interrupt is enabled.

REG[3834h] I2C Interrupt Raw Status Register
(Default = 0000h) Read Only

n/a					I2C Start Interrupt Raw Status	I2C Stop Interrupt Raw Status	I2C Busy Interrupt Raw Status
15	14	13	12	11	10	9	8
Reserved	I2C Transmit Abort Interrupt Raw Status	Reserved	I2C Transmit FIFO Empty Interrupt Raw Status	I2C Transmit FIFO Overflow Interrupt Raw Status	I2C Receive FIFO Full Interrupt Raw Status	I2C Receive FIFO Overflow Interrupt Raw Status	I2C Receive FIFO Underflow Interrupt Raw Status
7	6	5	4	3	2	1	0

bit 10 I2C Start Interrupt Raw Status (Read Only)
This bit indicates the raw status of the I2C Start Interrupt which is triggered when a start condition occurs on the I2C interface. This bit is not masked by the I2C Start Interrupt Enable bit, REG[3830h] bit 10. A masked status bit is available in REG[382Ch].
When this bit = 0b, an I2C Start Interrupt has not occurred.
When this bit = 1b, an I2C Start Interrupt has occurred.

To clear this status bit, read the I2C Start Interrupt Clear bit, REG[3864h] bit 0.

bit 9 I2C Stop Interrupt Raw Status (Read Only)
This bit indicates the raw status of the I2C Stop Interrupt which is triggered when a stop condition occurs on the I2C interface. This bit is not masked by the I2C Stop Interrupt Enable bit, REG[3830h] bit 9. A masked status bit is available in REG[382Ch].
When this bit = 0b, an I2C Stop Interrupt has not occurred.
When this bit = 1b, an I2C Stop Interrupt has occurred.

To clear this status bit, read the I2C Stop Interrupt Clear bit, REG[3860h] bit 0.

bit 8 I2C Busy Interrupt Raw Status (Read Only)
This bit indicates the raw status of the I2C Busy Interrupt which is triggered when there is activity on the I2C interface (I2C interface is busy). This bit remains set until cleared regardless of whether the I2C interface returns to an idle state. This bit is not masked by the I2C Busy Interrupt Enable bit, REG[3830h] bit 8. A masked status bit is available in REG[382Ch].
When this bit = 0b, an I2C Busy Interrupt has not occurred.
When this bit = 1b, an I2C Busy Interrupt has occurred.

To clear this status bit, read the I2C Busy Interrupt Clear bit, REG[385Ch] bit 0.

bit 7 Reserved
The default value for this bit is 0b.

bit 6 I2C Transmit Abort Interrupt Raw Status (Read Only)
This bit indicates the raw status of the I2C Transmit Abort Interrupt which triggers when the I2C acting as a master is unable to complete a command that the Host has sent. This bit is not masked by the I2C Transmit Abort Interrupt Enable bit, REG[3830h] bit 6. A masked status bit is available in REG[382Ch].
When this bit = 0b, an I2C Transmit Abort Interrupt has not occurred.
When this bit = 1b, an I2C Transmit Abort Interrupt has occurred.

To clear this status bit, read the I2C Transmit Abort Interrupt Clear bit, REG[3854h] bit 0.

The specific condition that caused the interrupt is indicated in the I2C Transmit Abort Source register (REG[3880h]) and identifies the following conditions.

- no slave acknowledges after the address is sent.
- the addressed slave does not acknowledge a byte of data.
- arbitration is lost.
- attempting to send a master command when configured only to be a slave.
- IC_RESTART_EN bit in the IC_CON register is set to 0 (restart condition disabled), and the processor attempts to issue an I2C function that is impossible to perform without using restart conditions.
- high-speed master code is acknowledge.
- start byte is acknowledged.
- general call address is not acknowledged.
- when a read request interrupt occurs and the processor has previously placed data in the TX buffer that has not been transmitted yet. This data could have been intended to service a multi-byte RD_REQ that ended up having fewer numbers of bytes requested. Or, if IC_RESTART_EN is disabled and the I2C loses control of the bus between transfers and is then accessed as a slave-transmitter.
- if a read command is issued after a general call command has been issued. Disabling the I2C reverts it back to normal operation.
- if the processor attempts to issue read command before a RD_REQ is serviced. Anytime this bit is set, the contents of the transmit and receive buffers are flushed.

bit 5 Reserved
The default value for this bit is 0b.

bit 4 I2C Transmit FIFO Empty Interrupt Raw Status (Read Only)
This bit indicates the raw status of the I2C Transmit FIFO Empty Interrupt which triggers when the Transmit FIFO level is at or below the level specified by the I2C Transmit FIFO Threshold Level bits (REG[383Ch] bits 2-0). This bit is not masked by the I2C Transmit FIFO Empty Interrupt Enable bit, REG[3830h] bit 4. A masked status bit is available in REG[382Ch].
When this bit = 0b, an I2C Transmit FIFO Empty Interrupt has not occurred.
When this bit = 1b, an I2C Transmit FIFO Empty Interrupt has occurred.

This status bit is cleared automatically when the Transmit FIFO level goes above the Transmit FIFO Threshold Level.
REG[3874h] bits 3-0 > REG[383Ch] bits 2-0

bit 3 I2C Transmit FIFO Overflow Interrupt Raw Status (Read Only)
This bit indicates the raw status of the I2C Transmit FIFO Overflow Interrupt which triggers when the Transmit FIFO is completely filled and the Host attempts to issue another I2C command by writing to REG[3810h]. This bit is not masked by the I2C Transmit FIFO Overflow Interrupt Enable bit, REG[3830h] bit 3. A masked status bit is available in REG[382Ch].
When this bit = 0b, an I2C Transmit FIFO Overflow Interrupt has not occurred.
When this bit = 1b, an I2C Transmit FIFO Overflow Interrupt has occurred.

To clear this status bit, read the I2C Transmit FIFO Overflow Interrupt Clear bit, REG[384Ch] bit 0.

bit 2 I2C Receive FIFO Full Interrupt Raw Status (Read Only)
This bit indicates the raw status of the I2C Receive FIFO Full Interrupt which triggers when the Receive FIFO level reaches or exceeds the level specified by the I2C Receive FIFO Threshold Level bits (REG[3838h] bits 2-0). This bit is not masked by the I2C Receive FIFO Full Interrupt Enable bit, REG[3830h] bit 2. A masked status bit is available in REG[382Ch].
When this bit = 0b, an I2C Receive FIFO Full Interrupt has not occurred.
When this bit = 1b, an I2C Receive FIFO Full Interrupt has occurred.

This status bit is cleared automatically when the Receive FIFO level goes below the Receive FIFO Threshold Level, REG[3878h] bits 3-0.
REG[3878h] bits 3-0 < REG[3838h] bits 2-0

bit 1 I2C Receive FIFO Overflow Interrupt Raw Status (Read Only)
This bit indicates the raw status of the I2C Receive FIFO Overflow Interrupt which is triggered when the Receive FIFO is completely full and more data arrives. When this happens, the data is lost. This bit is not masked by the I2C Receive FIFO Overflow Interrupt Enable bit, REG[3830h] bit 1. A masked status bit is available in REG[382Ch].
When this bit = 0b, an I2C Receive FIFO Overflow Interrupt has not occurred.
When this bit = 1b, an I2C Receive FIFO Overflow Interrupt has occurred.

To clear this status bit, read the I2C Receive FIFO Overflow Interrupt Clear bit, REG[3848h] bit 0.

bit 0 I2C Receive FIFO Underflow Interrupt Raw Status (Read Only)
This bit indicates the raw status of the I2C Receive FIFO Underflow Interrupt which is triggered when the Host attempts to read an empty Receive FIFO by reading from REG[3810h]. This bit is not masked by the I2C Receive FIFO Underflow Interrupt Enable bit, REG[3830h] bit 0. A masked status bit is available in REG[382Ch].
When this bit = 0b, an I2C Receive FIFO Underflow Interrupt has not occurred.
When this bit = 1b, an I2C Receive FIFO Underflow Interrupt has occurred.

To clear this status bit, read the I2C Receive FIFO Underflow Interrupt Clear bit, REG[3844h] bit 0.

REG[3838h] I2C Receive FIFO Threshold Register							
(Default = 0000h)							Read/Write
n/a							
15	14	13	12	11	10	9	8
n/a					I2C Receive FIFO Threshold Level bits 2-0		
7	6	5	4	3	2	1	0

bits 2-0 I2C Receive FIFO Threshold Level bits [2:0]
These bits control the threshold level (or number of entries) that will trigger the I2C Receive FIFO Full interrupt (see REG[382Ch] bit 2). The threshold level cannot be set larger than the depth of the Receive FIFO.
Receive FIFO Threshold = REG[3838h] bits 2-0 + 1

REG[383Ch] I2C Transmit FIFO Threshold Register							
(Default = 0000h)							Read/Write
n/a							
15	14	13	12	11	10	9	8
n/a					I2C Transmit FIFO Threshold Level bits 2-0		
7	6	5	4	3	2	1	0

bits 2-0 I2C Transmit FIFO Threshold Level bits [2:0]
These bits control the threshold level (or number of entries) that will trigger the I2C Transmit FIFO Empty interrupt (see REG[382Ch] bit 4). The threshold level cannot be set larger than the depth of the Transmit FIFO.
Transmit FIFO Threshold = REG[383Ch] bits 2-0

REG[3840h] I2C Clear Combined and Individual Interrupt Register							
(Default = 0000h)							Read Only
n/a							
15	14	13	12	11	10	9	8
n/a							I2C Interrupt Clear
7	6	5	4	3	2	1	0

bit 0 I2C Interrupt Clear (Read Only)
A read of this bit clears the combined interrupt, all individual interrupts (except for the I2C Transmit FIFO Empty and I2C Receive FIFO Full interrupts), and the I2C Transfer Abort Source register (REG[3880h]).

REG[3844h] I2C Receive FIFO Underflow Interrupt Clear Register (Default = 0000h)							Read Only
n/a							
15	14	13	12	11	10	9	8
n/a							I2C Receive FIFO Underflow Interrupt Clear
7	6	5	4	3	2	1	0

bit 0 I2C Receive FIFO Underflow Interrupt Clear (Read Only)
A read of this bit clears the I2C Receive FIFO Underflow Interrupt Status bit (REG[382Ch] bit 0) and the I2C Receive FIFO Underflow Interrupt Raw Status bit (REG[3834h] bit 0).

REG[3848h] I2C Receive FIFO Overflow Interrupt Clear Register (Default = 0000h)							Read Only
n/a							
15	14	13	12	11	10	9	8
n/a							I2C Receive FIFO Overflow Interrupt Clear
7	6	5	4	3	2	1	0

bit 0 I2C Receive FIFO Overflow Interrupt Clear (Read Only)
A read of this bit clears the I2C Receive FIFO Overflow Interrupt Status bit (REG[382Ch] bit 1) and the I2C Receive FIFO Overflow Interrupt Raw Status bit (REG[3834h] bit 1).

REG[384Ch] I2C Transmit FIFO Overflow Interrupt Clear Register Default = 0000h							Read Only
n/a							
15	14	13	12	11	10	9	8
n/a							I2C Transmit FIFO Overflow Interrupt Clear
7	6	5	4	3	2	1	0

bit 0 I2C Transmit FIFO Overflow Interrupt Clear (Read Only)
A read of this bit clears the I2C Transmit FIFO Overflow Interrupt Status bit (REG[382Ch] bit 3) and the I2C Transmit FIFO Overflow Interrupt Raw Status bit (REG[3834h] bit 3).

REG[3850h] is Reserved

This register is Reserved and should not be written.

REG[3854h] I2C Transmit Abort Interrupt Clear Register (Default = 0000h)							Read Only
n/a							
15	14	13	12	11	10	9	8
n/a							I2C Transmit Abort Interrupt Clear
7	6	5	4	3	2	1	0

bit 0 I2C Transmit Abort Interrupt Clear (Read Only)
A read of this bit clears the I2C Transmit Abort Interrupt Status bit (REG[382Ch] bit 6), the I2C Transmit Abort Interrupt Raw Status bit (REG[3834h] bit 6), and the I2C Transmit Abort Source register (REG[3880h]).

REG[3858h] is Reserved

This register is Reserved and should not be written.

REG[385Ch] I2C Busy Interrupt Clear Register (Default = 0000h)							Read Only
n/a							
15	14	13	12	11	10	9	8
n/a							I2C Busy Interrupt Clear
7	6	5	4	3	2	1	0

bit 0 I2C Busy Interrupt Clear (Read Only)
A read of this bit clears the I2C Busy Interrupt Status bit (REG[382Ch] bit 8) and the I2C Busy Interrupt Raw Status bit (REG[3834h] bit 8).

REG[3860h] I2C Stop Interrupt Clear Register (Default = 0000h)							Read Only
n/a							
15	14	13	12	11	10	9	8
n/a							I2C Stop Interrupt Clear
7	6	5	4	3	2	1	0

bit 0 I2C Stop Interrupt Clear (Read Only)
A read of this bit clears the I2C Stop Interrupt Status bit (REG[382Ch] bit 9) and the I2C Stop Interrupt Raw Status bit (REG[3834h] bit 9).

REG[3864h] I2C Start Interrupt Clear Register (Default = 0000h)							Read Only
n/a							
15	14	13	12	11	10	9	8
n/a							I2C Start Interrupt Clear
7	6	5	4	3	2	1	0

bit 0 I2C Start Interrupt Clear (Read Only)
A read of this bit clears the I2C Start Interrupt Status bit (REG[382Ch] bit 10) and the I2C Start Interrupt Raw Status bit (REG[3834h] bit 10).

REG[3868h] is Reserved

This register is Reserved and should not be written.

REG[386Ch] I2C Enable Register (Default = 0000h)							Read/Write
n/a							
15	14	13	12	11	10	9	8
n/a							I2C Enable
7	6	5	4	3	2	1	0

bit 0 I2C Enable
This bit controls whether the I2C module is enabled. The I2C module must not be disabled while it is still active (REG[3870h] bit 0 = 1b).
When this bit = 0b, the I2C module is disabled.
When this bit = 1b, the I2C module is enabled.

Note
When the I2C module is disabled, both the transmit and receive FIFOs will be cleared.

Note
To disable the I2C module after the last I2C command, use the following sequence:

1. Enable the I2C Busy Interrupt Status, write REG[3820h] bit 8 = 1b
2. Clear the I2C Busy Interrupt Status by reading REG[385Ch]
3. Write the last command to REG[3810h]
4. Wait for the I2C Busy Interrupt to trigger (either use the interrupt pin or check the I2C Busy Interrupt Status, REG[382Ch] bit 8)
5. Wait for the I2C Busy Status to show the I2C interface is idle, read REG[3870h] bit 0 = 0b
6. Disable the I2C module, write REG[386Ch] = 0000h
7. Clear the I2C Busy Interrupt Status by reading REG[385Ch]

REG[3870h] I2C Status Register (Default = 0006h)							Read Only
n/a							
15	14	13	12	11	10	9	8
n/a			Receive FIFO Full Status	Receive FIFO Not Empty Status	Transmit FIFO Empty Status	Transmit FIFO Not Full Status	I2C Busy Status
7	6	5	4	3	2	1	0

bit 4 Receive FIFO Full Status (Read Only)
This bit indicates the current "full status" of the Receive FIFO. When the Receive FIFO is completely full, this bit is set. When the Receive FIFO contains one or more empty location, this bit is cleared.
When this bit = 0b, the Receive FIFO is not full.
When this bit = 1b, the Receive FIFO is full.

bit 3 Receive FIFO Not Empty Status (Read Only)
This bit indicates the current “empty status” of the Receive FIFO. When the Receive FIFO contains one or more entries, this bit is set. When the Receive FIFO is completely empty, this bit is cleared. This bit can be polled by software to completely empty the Receive FIFO.
When this bit = 0b, the Receive FIFO is empty.
When this bit = 1b, the Receive FIFO is not empty.

bit 2 Transmit FIFO Empty Status (Read Only)
This bit indicates the current “empty status” of the Transmit FIFO. When the Transmit FIFO is completely empty, this bit is set. When the Transmit FIFO contains one or more valid entries, this bit is cleared. This bit field does not request an interrupt.
When this bit = 0b, the Transmit FIFO is not empty.
When this bit = 1b, the Transmit FIFO is empty.

bit 1 Transmit FIFO Not Full Status (Read Only)
This bit indicates the current “full status” of the Transmit FIFO. When the Transmit FIFO contains one or more empty locations, this bit is set. When the Transmit FIFO is full, this bit is cleared.
When this bit = 0b, the Transmit FIFO is full.
When this bit = 1b, the Transmit FIFO is not full.

bit 0 I2C Busy Status (Read Only)
This bit indicates the current “busy status” of the I2C interface.
When this bit = 0b, the I2C interface is idle.
When this bit = 1b, the I2C interface is busy.

REG[3874h] I2C Transmit FIFO Level Register (Default = 0000h)							Read Only
n/a							
15	14	13	12	11	10	9	8
n/a				I2C Transmit FIFO Level bits 3-0			
7	6	5	4	3	2	1	0

bits 3-0 I2C Transmit FIFO Level bits [3:0] (Read Only)
These bits indicate the current number of valid data entries in the I2C Transmit FIFO. The I2C Transmit FIFO is cleared when the I2C is disabled (REG[386Ch] bit 0 = 0b), a transmit abort occurs (REG[3834h] bit 6 = 1b), or when Slave Bulk Transfer mode is aborted. The FIFO level increments when data is placed into the Transmit FIFO and decrements when data is removed.

REG[3878h] I2C Receive FIFO Level Register
(Default = 0000h) Read Only

n/a							
15	14	13	12	11	10	9	8
n/a				I2C Receive FIFO Level bits 3-0			
7	6	5	4	3	2	1	0

bits 3-0 I2C Receive FIFO Level bits [3:0] (Read Only)
These bits indicate the current number of valid data entries in the I2C Receive FIFO. The I2C Receive FIFO is cleared when the I2C is disabled (REG[386Ch] bit 0 = 0b) or when a transmit abort occurs (REG[3834h] bit 6 = 1b). The FIFO level increments when data is placed into the Receive FIFO and decrements when data is removed.

REG[3880h] I2C Transmit Abort Source Register
(Default = 0000h) Read/Write

Slave Request Data to Transmit	Slave Arbitration Lost	Slave Flush Transmit FIFO	Arbitration Lost	Master Disabled	n/a		
15	14	13	12	11	10	9	8
Start Byte Acknowledged	n/a	General Call Read	General Call No ACK	Transmit Data No ACK	n/a		7-bit Address No ACK
7	6	5	4	3	2	1	0

Note

These bits are used to identify the cause (or source) of an I2C Transmit Abort Interrupt (see REG[382Ch] bit 6 or REG[3834h] bit 6). This register is cleared when the Host reads it, or when a 1b is written to the I2C Interrupt Clear bit (REG[3840h] bit 0).

bit 15 Slave Request Data to Transmit
This bit indicates the type of error that caused the Transmit Abort. This error is caused when the slave requests data to transmit and the user writes a read command into the Transmit FIFO (ninth bit is a 1b).
When this bit = 0b, this error did not cause the Transmit Abort.
When this bit = 1b, this error caused the Transmit Abort.

bit 14 Slave Arbitration Lost
This bit indicates the type of error that caused the Transmit Abort. This error is caused when the Slave looses the bus while transmitting data to a remote master. The Arbitration Lost bit is also set (REG[3880h] bit 12 = 1b) when this error occurs.
When this bit = 0b, this error did not cause the Transmit Abort.
When this bit = 1b, this error caused the Transmit Abort.

bit 13 Slave Flush Transmit FIFO
This bit indicates the type of error that caused the Transmit Abort. This error is caused when the Slave receives a read command and some data exists in the Transmit FIFO, so the slave issues a Transmit Abort to flush old data in the Transmit FIFO.
When this bit = 0b, this error did not cause the Transmit Abort.
When this bit = 1b, this error caused the Transmit Abort.

bit 12 Arbitration Lost
This bit indicates the type of error that caused the Transmit Abort. This error is caused when the Master loses arbitration, or if the Slave Arbitration Lost bit is also set (REG[3880h] bit 14 = 1b) the slave transmitter has lost arbitration.
When this bit = 0b, this error did not cause the Transmit Abort.
When this bit = 1b, this error caused the Transmit Abort.

bit 11 Master Disabled
This bit indicates the type of error that caused the Transmit Abort. This error is caused when the user attempts to disable the Master.
When this bit = 0b, this error did not cause the Transmit Abort.
When this bit = 1b, this error caused the Transmit Abort.

bit 7 Start Byte Acknowledged
This bit indicates the type of error that caused the Transmit Abort. This error is caused when the Master sends a start byte and the start byte is acknowledged (wrong behavior).
When this bit = 0b, this error did not cause the Transmit Abort.
When this bit = 1b, this error caused the Transmit Abort.

bit 5 General Call Read
This bit indicates the type of error that caused the Transmit Abort. This error is caused when the Master sends a general call, but the user programs the byte following the general call to be read from the bus (ninth bit is set to 1b).
When this bit = 0b, this error did not cause the Transmit Abort.
When this bit = 1b, this error caused the Transmit Abort.

bit 4 General Call No ACK
This bit indicates the type of error that caused the Transmit Abort. This error is caused when the Master sends a general call and no slave on the bus responds with an acknowledge.
When this bit = 0b, this error did not cause the Transmit Abort.
When this bit = 1b, this error caused the Transmit Abort.

bit 3 Transmit Data No ACK
This bit indicates the type of error that caused the Transmit Abort. This error is caused when the Master receives an acknowledgement for the address, but does not receive and acknowledge from the remote slave(s) when it sends the data byte(s) following the address.
When this bit = 0b, this error did not cause the Transmit Abort.
When this bit = 1b, this error caused the Transmit Abort.

bit 0 7-Bit Address No ACK
This bit indicates the type of error that caused the Transmit Abort. This error is caused when the Master is configured for 7-bit addressing mode and the address sent is not acknowledged by any slave.
When this bit = 0b, this error did not cause the Transmit Abort.
When this bit = 1b, this error caused the Transmit Abort.

REG[3888h] through REG[3890h] are Reserved

These registers are Reserved and should not be written.

REG[38F4h] through REG[38FEh] are Reserved

These registers are Reserved and should not be written.

10.4.18 DMA Control Registers

REG[3C00h] DMA Channel 0 Source Address Register 0
Default = not applicable Read/Write

DMA Channel 0 Source Address bits 15-8							
15	14	13	12	11	10	9	8
DMA Channel 0 Source Address bits 7-0							
7	6	5	4	3	2	1	0

REG[3C02h] DMA Channel 0 Source Address Register 1
Default = not applicable Read/Write

DMA Channel 0 Source Address bits 31-24							
15	14	13	12	11	10	9	8
DMA Channel 0 Source Address bits 23-16							
7	6	5	4	3	2	1	0

REG[3C02h] bits 15-0
REG[3C00h] bits 15-0 DMA Channel 0 Source Address bits [31:0]

These bits specify the 32-bit source address for DMA transfers on channel 0. This address must fall on a memory boundary that matches the transfer data size (see REG[3C0Ch] bits 4-3). For example, for 64-bit transfers, REG[3C00h] bits 2-0 must be 000b. After each successful transfer, this address is automatically updated according to the configuration of the DMA Channel 0 Source Address Mode bits, REG[3C0Ch] bits 13-12.

REG[3C04h] DMA Channel 0 Destination Address Register 0
Default = not applicable Read/Write

DMA Channel 0 Destination Address bits 15-8							
15	14	13	12	11	10	9	8
DMA Channel 0 Destination Address bits 7-0							
7	6	5	4	3	2	1	0

REG[3C06h] DMA Channel 0 Destination Address Register 1
Default = not applicable Read/Write

DMA Channel 0 Destination Address bits 31-24							
15	14	13	12	11	10	9	8
DMA Channel 0 Destination Address bits 23-16							
7	6	5	4	3	2	1	0

REG[3C06h] bits 15-0
REG[3C04h] bits 15-0 DMA Channel 0 Destination Address bits [31:0]

These bits specify the 32-bit destination address for DMA transfers on channel 0. This address must fall on a memory boundary that matches the transfer data size (see REG[3C0Ch] bits 4-3). For example, for 64-bit transfers, REG[3C04h] bits 2-0 must be 000b. After each successful transfer, this address is automatically updated according to the configuration of the DMA Channel 0 Destination Address Mode bits, REG[3C0Ch] bits 15-14.

REG[3C08h] DMA Channel 0 Transfer Count Register 0
Default = not applicable Write Only

DMA Channel 0 Transfer Count bits 15-8							
15	14	13	12	11	10	9	8
DMA Channel 0 Transfer Count bits 7-0							
7	6	5	4	3	2	1	0

REG[3C0Ah] DMA Channel 0 Transfer Count Register 1
Default = not applicable Write Only

n/a							
15	14	13	12	11	10	9	8
DMA Channel 0 Transfer Count bits 23-16							
7	6	5	4	3	2	1	0

REG[3C0Ah] bits 7-0
REG[3C08h] bits 15-0 DMA Channel 0 Transfer Count bits [23:0] (Write Only)

These bits specify the 24-bit transfer count for DMA transfers on channel 0. This count is automatically decremented after each successful DMA transfer. Starting at zero specifies 2^24 = 16,777,216 transfers. Decrementing to zero triggers a DMA interrupt request as an indicator of DMA transfer end.

REG[3C0Ch] DMA Channel 0 Control Register 0
Default = 0000 Read/Write

DMA Channel 0 Destination Address Mode bits 1-0		DMA Channel 0 Source Address Mode bits 1-0		DMA Channel 0 Resource bits 3-0			
15	14	13	12	11	10	9	8
Reserved	DMA Channel 0 Request Input Mode	Reserved	DMA Channel 0 Transfer Size bits 1-0		DMA Channel 0 Interrupt Request Enable	DMA Channel 0 Transfer End	DMA Channel 0 DMA Enable
7	6	5	4	3	2	1	0

bits 15-14 DMA Channel 0 Destination Address Mode bits [1:0]

These bits select the method used to update the DMA Channel 0 Destination Address registers (REG[3C04h] ~ REG[3C06h]) after a successful DMA transfer.

Table 10-108 : DMA Channel 0 Destination Address Mode Selection

REG[3C0Ch] bits 15-14	DMA Channel 0 Destination Address Mode
00b	Leave fixed (Do not update)
01b	Increment according to the transfer data size (see REG[3C0Ch] bits 4-3) (8-bit: +1, 16-bit: +2, 32-bit: +4, 64-bit: +8)
10b	Decrement according to the transfer data size (see REG[3C0Ch] bits 4-3) (8-bit: -1, 16-bit: -2, 32-bit: -4, 64-bit: -8)
11b	Reserved

bits 13-12 DMA Channel 0 Source Address Mode bits [1:0]
These bits select the method used to update the DMA Channel 0 Source Address registers (REG[3C00h] ~ REG[3C02h]) after a successful DMA transfer.

Table 10-109 : DMA Channel 0 Source Address Mode Selection

REG[3C0Ch] bits 13-12	DMA Channel 0 Source Address Mode
00b	Leave fixed (Do not update)
01b	Increment according to the transfer data size (see REG[3C0Ch] bits 4-3) (8-bit: +1, 16-bit: +2, 32-bit: +4, 64-bit: +8)
10b	Decrement according to the transfer data size (see REG[3C0Ch] bits 4-3) (8-bit: -1, 16-bit: -2, 32-bit: -4, 64-bit: -8)
11b	Reserved

bits 11-8 DMA Channel 0 Resource bits [3:0]
These bits determine which IO device (or resource) is assigned to DMA Channel 0.

Table 10-110 : DMA Channel 0 Resource Map

REG[3C0Ch] bits 11-8	DMA Channel 0 Resource
0000b	YUV Data Capture Request
0001b	Memory to YRC (YUV to RGB Converter) Request
0010b	Reserved
0011b	Reserved
0100b - 1110b	Reserved
1111b	Software Request (SW-Request)

bit 7 Reserved
The default value for this bit is 0b.

bit 6 DMA Channel 0 Request Input Mode
This bit specifies the input mode for the DMA Channel 0 DMA request signal from the resource specified by REG[3C0Ch] bits 11-8.
When this bit = 0b, the input mode is active low (level trigger).
When this bit = 1b, the input mode is falling edge (edge trigger).

bit 5 Reserved
This bit must be set to 0b.

bits 4-3 DMA Channel 0 Transfer Size bits [1:0]
These bits select the transfer size for DMA Channel 0.

Table 10-111 : DMA Channel 0 Transfer Size Selection

REG[3C0Ch] bits 4-3	DMA Channel 0 Transfer Size
00b	8-bit
01b	16-bit
10b	32-bit
11b	64-bit

bit 2 DMA Channel 0 Interrupt Request Enable
This bit controls whether an interrupt request is generated after a transfer on DMA Channel 0 completes. If enabled, the interrupt occurs when the DMA Channel 0 Transfer Count (see REG[3C08h] ~ REG[3C0Ah]) reaches zero.
When this bit = 0b, the interrupt request is disabled.
When this bit = 1b, the interrupt request is enabled.

bit 1 DMA Channel 0 Transfer End
This bit indicates when all transfers on DMA Channel 0 are complete which occurs when the DMA Channel 0 Transfer Count (see REG[3C08h] ~ REG[3C0Ah]) decrements to zero. DMA transfers on DMA Channel 0 are disabled until this bit is cleared by software. This bit can also be used as the DMA Channel 0 Interrupt Request source flag.
When this bit = 0b, transfers are in progress on DMA Channel 0 or the channel is idle.
When this bit = 1b, the transfer on DMA Channel 0 is complete.

To clear this bit, write a 0b to this bit.

bit 0 DMA Channel 0 Enable
This bit controls transfers on DMA Channel 0.
For Reads:
When this bit = 0b, transfers are disabled.
When this bit = 1b, a transfer has been triggered.
For Writes:
Writing a 0b to this bit has no effect.
Writing a 1b to this bit when REG[3C0Ch] bit 1 = 0b will trigger a DMA transfer.

REG[3C0Eh] DMA Channel 0 Control Register 1
Default = 0000 Read/Write

n/a							
15	14	13	12	11	10	9	8
Reserved				DMA Channel 0 Idle Delay Enable	Reserved	DMA Channel 0 Acknowledge Mode	DMA Channel 0 Acknowledge Level
7	6	5	4	3	2	1	0

bits 7-4 Reserved
The default value for these bits is 0000b.

bit 3 DMA Channel 0 Idle Delay Enable
This bit controls the DMA Channel 0 idle delay which enables a delay in accepting the next request from the device. This may be required for some target devices. For example, it is recommended to set this bit to 1b for write transfers from memory to IO devices (IO write transfers).
When this bit = 0b, the idle delay is disabled (normal operation).
When this bit = 1b, the idle delay is enabled.

bit 2 Reserved
The default value for this bit is 0b.

bit 1 DMA Channel 0 Acknowledge Mode
This bit selects the DACK signal output active timing.
When this bit = 0b, the DACK signal is active in the DMA read cycle.
When this bit = 1b, the DACK signal is active in the DMA write cycle.

bit 0 DMA Channel 0 Acknowledge Level
This bit selects the DACK signal output polarity.
When this bit = 0b, the DACK signal output polarity is active low.
When this bit = 1b, the DACK signal output polarity is active high.

REG[3C10h] DMA Channel 1 Source Address Register 0 Default = not applicable							Read/Write
DMA Channel 1 Source Address bits 15-8							
15	14	13	12	11	10	9	8
DMA Channel 1 Source Address bits 7-0							
7	6	5	4	3	2	1	0

REG[3C12h] DMA Channel 1 Source Address Register 1 Default = not applicable							Read/Write
DMA Channel 1 Source Address bits 31-24							
15	14	13	12	11	10	9	8
DMA Channel 1 Source Address bits 23-16							
7	6	5	4	3	2	1	0

REG[3C12h] bits 15-0
REG[3C10h] bits 15-0 DMA Channel 1 Source Address bits [31:0]
These bits specify the 32-bit source address for DMA transfers on channel 1. This address must fall on a memory boundary that matches the transfer data size (see REG[3C1Ch] bits 4-3). For example, for 64-bit transfers, REG[3C10h] bits 2-0 must be 000b. After each successful transfer, this address is automatically updated according to the configuration of the DMA Channel 1 Source Address Mode bits, REG[3C1Ch] bits 13-12.

REG[3C14h] DMA Channel 1 Destination Address Register 0							
Default = not applicable							Read/Write
DMA Channel 1 Destination Address bits 15-8							
15	14	13	12	11	10	9	8
DMA Channel 1 Destination Address bits 7-0							
7	6	5	4	3	2	1	0

REG[3C16h] DMA Channel 1 Destination Address Register 1							
Default = not applicable							Read/Write
DMA Channel 1 Destination Address bits 31-24							
15	14	13	12	11	10	9	8
DMA Channel 1 Destination Address bits 23-16							
7	6	5	4	3	2	1	0

REG[3C16h] bits 15-0
REG[3C14h] bits 15-0 DMA Channel 1 Destination Address bits [31:0]

These bits specify the 32-bit destination address for DMA transfers on channel 1. This address must fall on a memory boundary that matches the transfer data size (see REG[3C1Ch] bits 4-3). For example, for 64-bit transfers, REG[3C14h] bits 2-0 must be 000b. After each successful transfer, this address is automatically updated according to the configuration of the DMA Channel 1 Destination Address Mode bits, REG[3C1Ch] bits 15-14.

REG[3C18h] DMA Channel 1 Transfer Count Register 0							
Default = not applicable							Write Only
DMA Channel 1 Transfer Count bits 15-8							
15	14	13	12	11	10	9	8
DMA Channel 1 Transfer Count bits 7-0							
7	6	5	4	3	2	1	0

REG[3C1Ah] DMA Channel 1 Transfer Count Register 1							
Default = not applicable							Write Only
n/a							
15	14	13	12	11	10	9	8
DMA Channel 1 Transfer Count bits 23-16							
7	6	5	4	3	2	1	0

REG[3C1Ah] bits 7-0
REG[3C18h] bits 15-0 DMA Channel 1 Transfer Count bits [23:0] (Write Only)

These bits specify the 24-bit transfer count for DMA transfers on channel 1. This count is automatically decremented after each successful DMA transfer. Starting at zero specifies 2^24 = 16,777,216 transfers. Decrementing to zero triggers a DMA interrupt request as an indicator of DMA transfer end.

REG[3C1Ch] DMA Channel 1 Control Register 0
Default = 0000 Read/Write

DMA Channel 1 Destination Address Mode bits 1-0		DMA Channel 1 Source Address Mode bits 1-0		DMA Channel 1 Resource bits 3-0			
15	14	13	12	11	10	9	8
Reserved	DMA Channel 1 Request Input Mode	Reserved	DMA Channel 1 Transfer Size bits 1-0		DMA Channel 1 Interrupt Request Enable	DMA Channel 1 Transfer End	DMA Channel 1 DMA Enable
7	6	5	4	3	2	1	0

bits 15-14 DMA Channel 1 Destination Address Mode bits [1:0]
These bits select the method used to update the DMA Channel 1 Destination Address registers (REG[3C14h] ~ REG[3C16h]) after a successful DMA transfer.

Table 10-112 : DMA Channel 1 Destination Address Mode Selection

REG[3C1Ch] bits 15-14	DMA Channel 1 Destination Address Mode
00b	Leave fixed (Do not update)
01b	Increment according to the transfer data size (see REG[3C1Ch] bits 4-3) (8-bit: +1, 16-bit: +2, 32-bit: +4, 64-bit: +8)
10b	Decrement according to the transfer data size (see REG[3C1Ch] bits 4-3) (8-bit: -1, 16-bit: -2, 32-bit: -4, 64-bit: -8)
11b	Reserved

bits 13-12 DMA Channel 1 Source Address Mode bits [1:0]
These bits select the method used to update the DMA Channel 1 Source Address registers (REG[3C10h] ~ REG[3C12h]) after a successful DMA transfer.

Table 10-113 : DMA Channel 1 Source Address Mode Selection

REG[3C1Ch] bits 13-12	DMA Channel 1 Source Address Mode
00b	Leave fixed (Do not update)
01b	Increment according to the transfer data size (see REG[3C1Ch] bits 4-3) (8-bit: +1, 16-bit: +2, 32-bit: +4, 64-bit: +8)
10b	Decrement according to the transfer data size (see REG[3C1Ch] bits 4-3) (8-bit: -1, 16-bit: -2, 32-bit: -4, 64-bit: -8)
11b	Reserved

bits 11-8 DMA Channel 1 Resource bits [3:0]
These bits determine which IO device (or resource) is assigned to DMA Channel 1.

Table 10-114 : DMA Channel 1 Resource Map

REG[3C1Ch] bits 11-8	DMA Channel 1 Resource
0000b	YUV Data Capture Request
0001b	Memory to YRC (YUV to RGB Converter) Request
0010b	Reserved
0011b	Reserved
0100b - 1110b	Reserved
1111b	Software request (SW-Request)

bit 7 Reserved
The default value for this bit is 0b.

bit 6 DMA Channel 1 Request Input Mode
This bit specifies the input mode for the DMA Channel 1 DMA request signal from the resource specified by REG[3C1Ch] bits 11-8.
When this bit = 0b, the input mode is active low (level trigger).
When this bit = 1b, the input mode is falling edge (edge trigger).

bit 5 Reserved
This bit must be set to 0b.

bits 4-3 DMA Channel 1 Transfer Size bits [1:0]
These bits select the transfer size for DMA Channel 1.

Table 10-115 : DMA Channel 1 Transfer Size Selection

REG[3C1Ch] bits 4-3	DMA Channel 1 Transfer Size
00b	8-bit
01b	16-bit
10b	32-bit
11b	64-bit

bit 2 DMA Channel 1 Interrupt Request Enable
This bit controls whether an interrupt request is generated after a transfer on DMA Channel 1 completes. If enabled, the interrupt occurs when the DMA Channel 1 Transfer Count (see REG[3C18h] ~ REG[3C1Ah]) reaches zero.
When this bit = 0b, the interrupt request is disabled.
When this bit = 1b, the interrupt request is enabled.

bit 1 DMA Channel 1 Transfer End
This bit indicates when all transfers on DMA Channel 1 are complete which occurs when the DMA Channel 1 Transfer Count (see REG[3C18h] ~ REG[3C1Ah]) decrements to zero. DMA transfers on DMA Channel 1 are disabled until this bit is cleared by software. This bit can also be used as the DMA Channel 1 Interrupt Request source flag.
When this bit = 0b, transfers are in progress on DMA Channel 1 or the channel is idle.
When this bit = 1b, the transfer on DMA Channel 1 is complete.

To clear this bit, write a 0b to this bit.

bit 0 DMA Channel 1 Enable
This bit controls transfers on DMA Channel 1.
For Reads:
When this bit = 0b, transfers are disabled.
When this bit = 1b, a transfer has been triggered.
For Writes:
Writing 0b to this bit has no effect.
Writing 1b to this bit when REG[3C1Ch] bit 1 = 0b will trigger a DMA transfer.

REG[3C1Eh] DMA Channel 1 Control Register 1
Default = 0000 Read/Write

n/a							
15	14	13	12	11	10	9	8
Reserved				DMA Channel 1 Idle Delay Enable	Reserved	DMA Channel 1 Acknowledge Mode	DMA Channel 1 Acknowledge Level
7	6	5	4	3	2	1	0

bits 7-4 Reserved
The default value for these bits is 0000b.

bit 3 DMA Channel 1 Idle Delay Enable
This bit controls the DMA Channel 1 idle delay which enables a delay in accepting the next request from the device. This may be required for some target devices. For example, it is recommended to set this bit to 1b for write transfers from memory to IO devices (IO write transfers).
When this bit = 0b, the idle delay is disabled (normal operation).
When this bit = 1b, the idle delay is enabled.

bit 2 Reserved
The default value for this bit is 0b.

bit 1 DMA Channel 1 Acknowledge Mode
This bit selects the DACK signal output active timing.
When this bit = 0b, the DACK signal is active in the DMA read cycle.
When this bit = 1b, the DACK signal is active in the DMA write cycle.

bit 0 DMA Channel 1 Acknowledge Level
This bit selects the DACK signal output polarity.
When this bit = 0b, the DACK signal output polarity is active low.
When this bit = 1b, the DACK signal output polarity is active high.

REG[3C60h] DMA Channel Operating Select Register Default = 0000							Read/Write
n/a						DMA Priority Toggling Enable	DMA Priority Mode
15	14	13	12	11	10	9	8
n/a							DMA Global Enable
7	6	5	4	3	2	1	0

bit 9 DMA Priority Toggling Enable
This bit determines whether the DMA Priority Mode bit is toggled.
When this bit = 0b, priority toggling is disabled.
When this bit = 1b, priority toggling is enabled.

bit 8 DMA Priority Mode
This bit selects which DMA channel is high priority.
When this bit = 0b, DMA Channel 0 is high priority.
When this bit = 1b, DMA Channel 1 is high priority.

bit 0 DMA Global Enable
This bit is the global control for both DMA Channels.
When this bit = 0b, both DMA Channel 0 and DMA Channel 1 are disabled.
When this bit = 1b, both DMA Channel 0 and DMA Channel 1 are enabled.

Note

This bit should be enabled (REG[3C60h] bit 0 = 1b) before any DMA transfers are triggered.

REG[3C64h] DMA Channel Miscellaneous Register Default = 0000							Read/Write
DMAC Software Reset	n/a						
15	14	13	12	11	10	9	8
n/a						DMA Channel 1 Polarity	DMA Channel 0 Polarity
7	6	5	4	3	2	1	0

bit 15 DMAC Software Reset
This bit performs a software reset of the DMA controller and resets all DMA Control registers (REG[3C00h] ~ REG[3C70h]) to their default values.
Writing a 0b to this bit has no hardware effect.
Writing a 1b to this bit performs a software reset of the DMA controller.

bit 1 DMA Channel 1 Polarity
This bit selects the polarity of DMA Channel 1.
When this bit = 0b, the polarity of DMA Channel 1 is positive.
When this bit = 1b, the polarity of DMA Channel 1 is negative.

bit 0 DMA Channel 0 Polarity
This bit selects the polarity of DMA Channel 0.
When this bit = 0b, the polarity of DMA Channel 0 is positive.
When this bit = 1b, the polarity of DMA Channel 0 is negative.

REG[3C70h] DMA Channel Transfer Complete Control Register
Default = 0000 Read/Write

n/a		TE Set Enable Upon Transfer Complete Acceptance	Transfer Complete Acceptance Enable	n/a			
15	14	13	12	11	10	9	8
n/a							
7	6	5	4	3	2	1	0

bit 13 TE Set Enable Upon Transfer Complete Acceptance (STTE)
When this bit = 0b, Disable
When this bit = 1b, Enable
Writing 1b to this bit accepts of the transfer complete signal from the interrupt request source and sets the transfer end (TE) bit to 1b.
This bit is valid only when bit 12 (ENTE) is set to 1b.

bit 12 Transfer Complete Acceptance Enable (ENTE)
When this bit = 0b, Disable
When this bit = 1b, Enable
Writing 1b to this bit accepts the transfer complete signal from the interrupt request source without setting the transfer end (TE) bit to 1b.

Note
Setting ENTE to 1b requires that a 1b is written to bit 13 (STTE).

10.4.19 Command FIFO (For BitBLT and Sprite) Registers

The Command FIFO does not have any dedicated register space. However, the register space starting at REG[4000h] is used to enter commands to the Command FIFO. When data is written to a register in this space, it is transferred into the Command FIFO to await processing by either the Sprite Engine or the BitBLT engine.

Reading any register within the Command FIFO register space (REG[4000h] ~ REG[4FFFh]) will return the number of entries currently available in the Command FIFO. When the Command FIFO is empty, a read will return 40h (64). When the Command FIFO is being used, it will return (64 - the number of used entries).

The Command FIFO space is mapped using the same offset addresses as the Sprite registers (REG[1000h] ~ REG[17FFh]) and BitBLT registers (REG[1800h] ~ REG[1FFFh]). For example, writing a value to REG[4700h] will load the Command FIFO with a command to program the Sprite Control register (REG[1700h]) with the specified value. The Command FIFO register space if mapped as follows.

Table 10-116 : Command FIFO Mapping

Command FIFO Address Range	Space Usage	Corresponding Sprite/BitBLT Register
REG[4000h] ~ REG[47FFh]	Sprite IO Space for Command FIFO	REG[1000h] ~ REG[17FFh]
REG[4800h] ~ REG[4FFFh]	BitBLT IO Space for Command FIFO	REG[1800h] ~ REG[1FFFh]

For further information on the Command FIFO, see Section 17, “Command FIFO” on page 424.

Note

The clock for the Command FIFO (REG[0462h] bit 5) must be enabled before any command write or read access to the Command FIFO.

10.4.20 Keypad Interface Register

The Keypad Interface supports input from up to a 5 x 5 matrix keypad. For keypad connection details and programming examples, refer to Section 24, “Keypad Interface” on page 476.

All keypad function pins are multiplexed on GPIO function pins. Before using the keypad interface, they must be re-programmed as keypad interface pins by setting the alternate function in the GPIO registers, REG[0C00h] ~ REG[0C1Eh]. For a summary of GPIO pin usage, see Section 5.6, “GPIO Pin Mapping” on page 50.

Note
The keypad interface is not available for the QFP package.

REG[5000h] Key Control Register
Default = 0000h Read/Write

Keypad Global Interrupt Enable	n/a						
15	14	13	12	11	10	9	8
n/a						Keypad Drive Output Polarity Select	Keypad Interface Enable
7	6	5	4	3	2	1	0

bit 15 Keypad Global Interrupt Enable
This bit controls the global interrupt for the keypad interface.
When this bit = 0b, the global interrupt is disabled. (default)
When this bit = 1b, the global interrupt is enabled.

bit 1 Keypad Drive Output Polarity Select
This bit selects the polarity of the drive output for the keypad interface.
When this bit = 0b, drive output is normal (Low Drive). (default)
When this bit = 1b, drive output is reversed (High Drive).

bit 0 Keypad Interface Enable
This bit controls the keypad interface and determines whether the keys are sampled.
When this bit = 0b, the keypad interface is disabled. (default)
When this bit = 1b, the keypad interface is enabled and the keys are sampled.

Note
Key scan output will start as soon as the key pad interface is enabled.

REG[5002h] Key Interface Interrupt Status Register

Default = 0000h Read Only

Keypad Global Interrupt Raw Status	n/a						
15	14	13	12	11	10	9	8
n/a			Key Scan Input Data 4 Raw Interrupt Status	Key Scan Input Data 3 Raw Interrupt Status	Key Scan Input Data 2 Raw Interrupt Status	Key Scan Input Data 1 Raw Interrupt Status	Key Scan Input Data 0 Raw Interrupt Status
7	6	5	4	3	2	1	0

bit 15 Keypad Global Interrupt Raw Status (Read Only)
This bit indicates the raw status of the global interrupt for the keypad interface.
When this bit = 0b, a global interrupt has not occurred.
When this bit = 1b, a global interrupt has occurred.

bit 4 Key Scan Input Data 4 Raw Interrupt Status
When this bit = 0b, an interrupt has not occurred (No IRQ).
When this bit = 1b, an interrupt has occurred (IRQ Source bit).

bit 3 Key Scan Input Data 3 Raw Interrupt Status
When this bit = 0b, an interrupt has not occurred (No IRQ).
When this bit = 1b, an interrupt has occurred (IRQ Source bit).

bit 2 Key Scan Input Data 2 Raw Interrupt Status
When this bit = 0b, an interrupt has not occurred (No IRQ).
When this bit = 1b, an interrupt has occurred (IRQ Source bit).

bit 1 Key Scan Input Data 1 Raw Interrupt Status
When this bit = 0b, an interrupt has not occurred (No IRQ).
When this bit = 1b, an interrupt has occurred (IRQ Source bit).

bit 0 Key Scan Input Data 0 Raw Interrupt Status
When this bit = 0b, an interrupt has not occurred (No IRQ).
When this bit = 1b, an interrupt has occurred (IRQ Source bit).

Note

1. Once these bits indicate an interrupt has occurred (bit = 1b), they are cleared once a 1b is written to the Key Scan Out Start bit (REG[500Eh] bit 0).
2. Once a keypress of one of the 5 x 5 matrix is detected, the scan output stops driving. To re-enable scan output, write a 1b to the Key Scan Out Start bit (REG[500Eh] bit 0).

REG[5004h] Key Scan Data Register
Default = 1E00h Read Only

n/a			Key Scan Data Output bits 4-0				
15	14	13	12	11	10	9	8
n/a			Key Scan Data Input bits 4-0				
7	6	5	4	3	2	1	0

bits 12-8 Key Scan Data Output bits [4:0] (Read Only)
When the corresponding bit in REG[5002h] register is 1b, key scan data is available by reading these bits.
To clear the Interrupt Status bits, restart the scan out by setting REG[500Eh] bit 0 to 1b.

bits 4-0 Key Scan Data Input bits [4:0]
When the corresponding bit in REG[5002h] register is 1b, key scan data is available by reading these bits.
To clear the Interrupt Status bits, restart the scan out by setting REG[500Eh] bit 0 to 1b.

REG[5006h] Key Scan Input Filter Clock Register
Default = 0000h Read/Write

Key Scan Filter Clock Frequency bits 15-8							
15	14	13	12	11	10	9	8
Key Scan Filter Clock Frequency bits 7-0							
7	6	5	4	3	2	1	0

bits 15-0 Key Scan Filter Clock Frequency bits [15:0]
These bits specify the key scan filter clock frequency for each keypad input pin when the key scan input filter is enabled, REG[5008h] bit 15 = 1b. When the key scan input filter is enabled, the key scan filter clock frequency must be at least 3 times greater than the key scan output driving frequency. (i.e. REG[500Ah] + 1 > (REG[5006h] + 1) x 3).

When the key scan input filter is enabled, the key scan input data must sample the same value three times consecutively in order for the key scan input to change to that value.
Key Scan Filter Clock Frequency
= SYSCLK ÷ (REG[042Ch] bits 11-0 + 1) ÷ (REG[5006h] bits 15-0 + 1)

REG[5008h] Key GPI Control Register
Default = 0000h Read/Write

Key Scan Input Filter Enable	n/a						
15	14	13	12	11	10	9	8
n/a			Key Scan Input Polarity Select bits 4-0				
7	6	5	4	3	2	1	0

bit 15 Key Scan Input Filter Enable
This bit controls the key scan input filter.
When this bit = 0b, the input filter is disabled.
When this bit = 1b, the input filter is enabled.

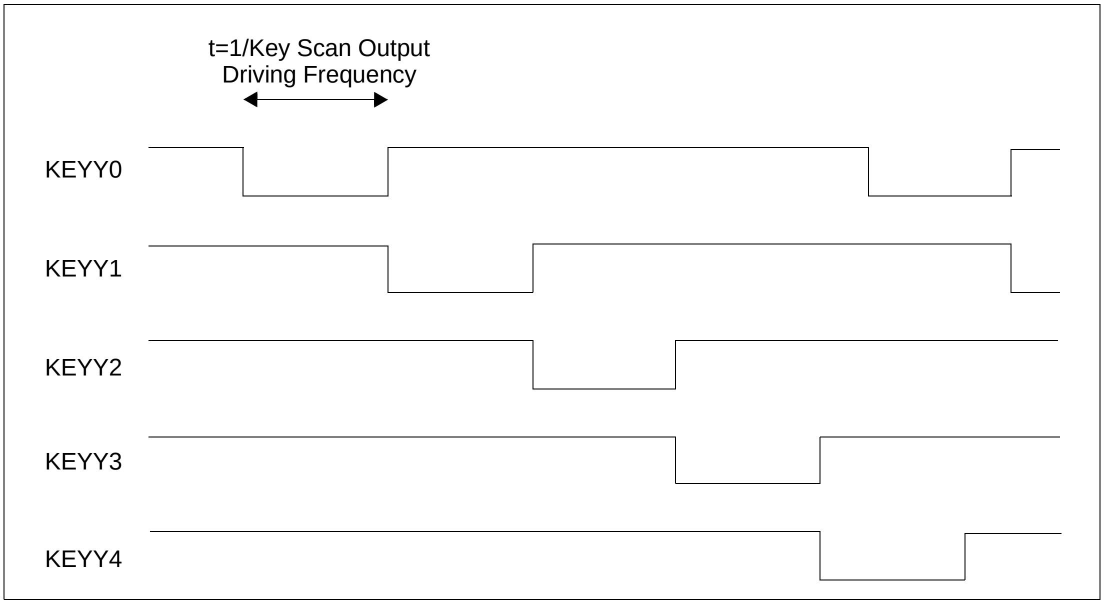
bits 4-0 Key Scan Input Polarity Select bits [4:0]
When this bit = 0b, the input polarity is normal.
When this bit = 1b, the input polarity is reversed (through one level of inverter).

Note

For high key pad drive polarity (REG[5000h] bit 1 = 1b), these bits must be set to 1Fh.

REG[500Ah] Key Scan Output Control Register Default = 0000h							Read/Write
Key Scan Output Driving Frequency bits 15-8							
15	14	13	12	11	10	9	8
Key Scan Output Driving Frequency bits 7-0							
7	6	5	4	3	2	1	0

bits 15-0 Key Scan Output Driving Frequency bits [15:0]
These bits specify the key scan output driving frequency for each keypad output pin. When the key scan input filter is enabled, the key scan filter clock frequency must be at least 3 times greater than the key scan output driving frequency. (i.e. REG[500Ah] + 1 > (REG[5006h] + 1) x 3).
Key Scan Output Driving Frequency
= SYSCLK ÷ (REG[042Ch] bits 11-0 + 1) ÷ (REG[500Ah] bits 15-0 + 1)



REG[500Ch] Key Scan GPI Filtered Register
Default = XX1Fh Read/Write

n/a			Raw GPI Input bits 4-0				
15	14	13	12	11	10	9	8
n/a			Filtered GPI Input bits 4-0				
7	6	5	4	3	2	1	0

bits 12-8 Raw GPI Input bits [4:0]
These bits indicate the raw state of the GPIs (General Purpose Input) that are used for Key Scan input (GPIOA[4:0]).

bits 4-0 Filtered GPI Input bits [4:0]
These bits indicate the filtered state of the GPIs (General Purpose Input) that are used for Key Scan input.

REG[500Eh] Key Scan Re-Enable Register
Default = 1E00h Read/Write

n/a			Reserved				
15	14	13	12	11	10	9	8
n/a							Key Scan Out Restart (WO)
7	6	5	4	3	2	1	0

bits 12-8 Reserved
The default value for these bits is 1_1110b.

bit 0 Key Scan Out Restart (Write Only)
This bit restarts the scan out function for the keypad interface. The scan out function is disabled when the keypad interface detects an input key being pressed. To detect new key presses, set this bit to 1b which will restart the scan out function.
Writing a 0b to this bit has no hardware effect.
Writing a 1b to this bit restarts the scanning output.

REG[5010h] Key Scan Active High Drive Register
Default = 0100h Read/Write

n/a			Inverted Scan Out Value bits 4-0				
15	14	13	12	11	10	9	8
n/a			Key Scan Data Input bits 4-0 (Active High)				
7	6	5	4	3	2	1	0

bits 12-8 Inverted Scan Out Value bits [4:0] (Read Only)
These bits determine the inverted value of the scan output (NOT of the scan out pin state = active high value).

bits 4-0 Key Scan Data Input bits [4:0]
When the corresponding bit in REG[5002h] register is 1b, then key scan data is available by reading these bits.
To clear the Interrupt Status bits, restart the scan out by setting REG[500Eh] bit 0 to 1b.

11 Power Save Modes

11.1 Power-On/Power-Off Sequence

The following power-on/power-off sequence is recommended for the S1D13513.

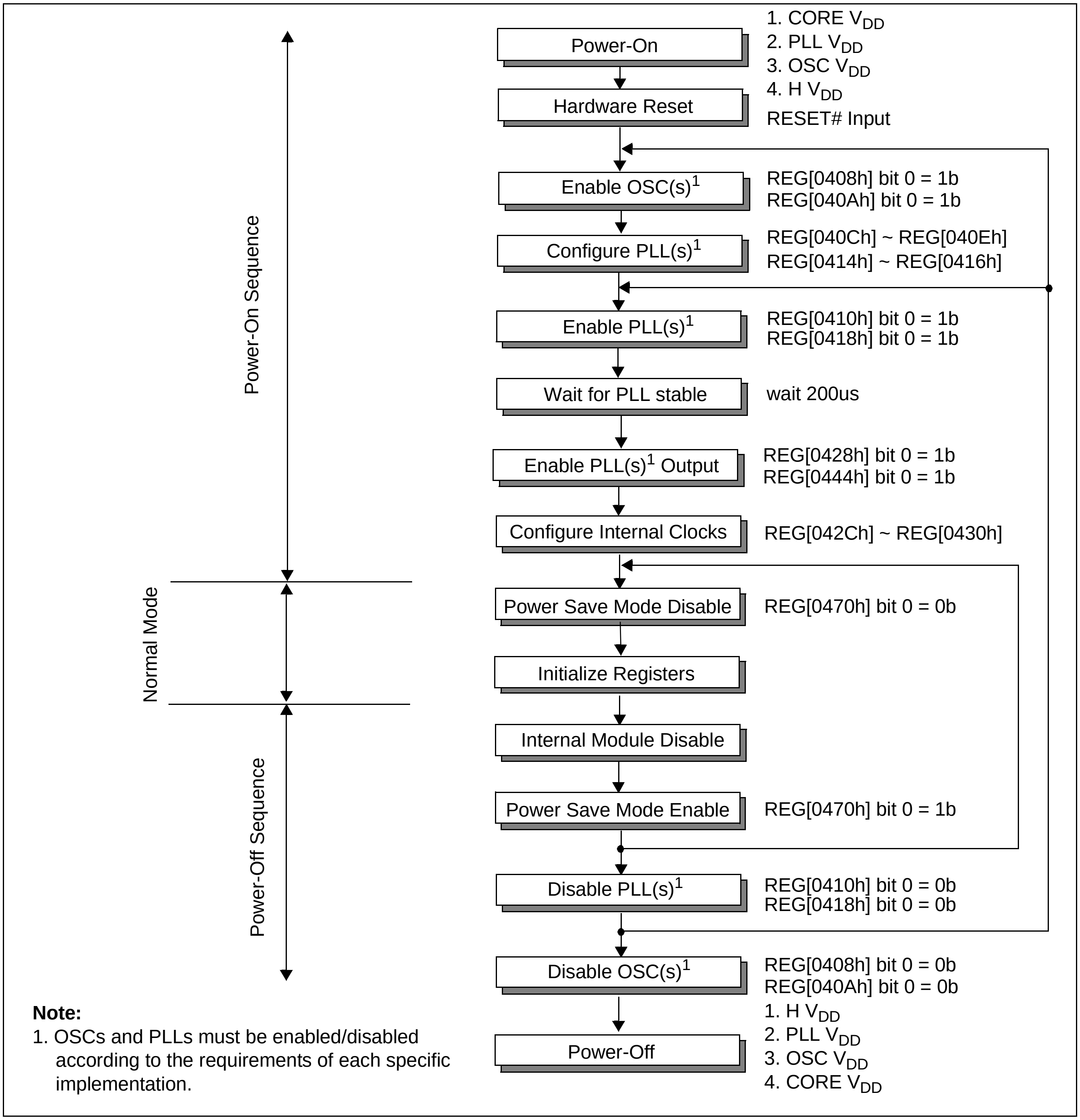


Figure 11-1: Power-On/Power-Off Sequence

11.2 Operational Modes

The S1D13513 operates in the following modes.

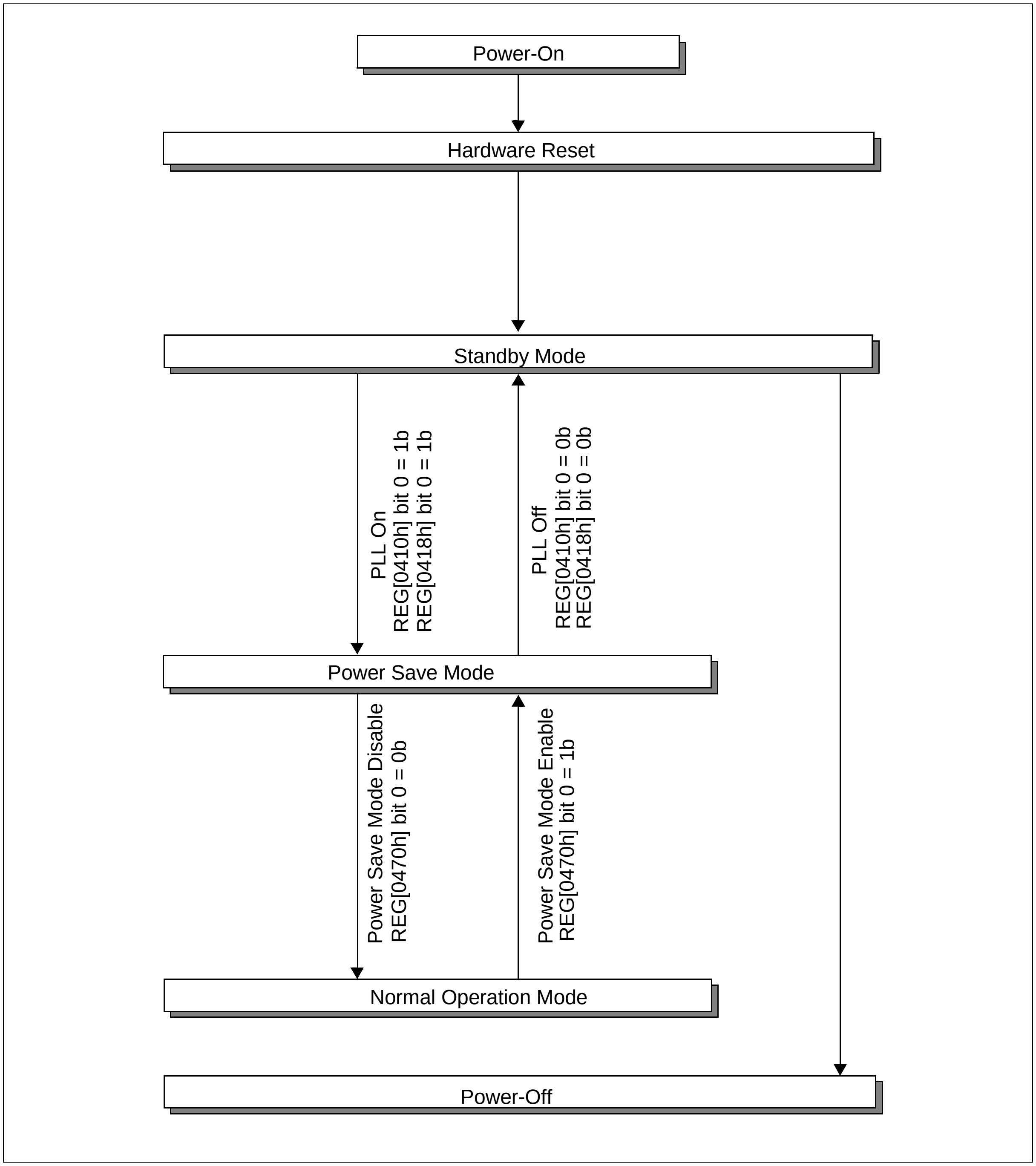


Figure 11-2: Operational Modes

11.2.1 Power-On

When powering-on the S1D13513, the following sequence must be used.

1. CORE V_{DD} On, OSC V_{DD}, PLL V_{DD} On
2. H V_{DD} (HVDD1 ~ HVDD5) On

Note
H V_{DD} is 0.8V when CORE V_{DD} is enabled and H V_{DD} is not enabled.

11.2.2 Reset

After power-on, an active low hardware reset pulse, which is two external clock cycles (see Power-On Section and RESET# section???) in length, must be input to the S1D13513 RESET# pin. All registers are reset by a hardware reset. After releasing the RESET# signal, the Clock Setting registers are immediately accessible.

A software reset is enabled by writing a value of A55Ah to REG[0460h]. All synchronous registers are reset to their default values (see Section 10.1, “Register Mapping” on page 126).

11.2.3 Standby Mode

Standby Mode offers the lowest power consumption because all internal clock supplies are stopped and the PLL is disabled. This mode must be entered before turning off the power supplies or setting the PLL registers.

In standby mode, the asynchronous registers (see Section 10.1, “Register Mapping” on page 126) can be accessed.

11.2.4 Power Save Mode

Power Save Mode stops all internal clock supplies. This mode must be entered before configuring the internal clocks (REG[042Ch] ~ REG[0430h]). Also, there may be up to a 200us delay before the PLL output becomes stable after it is enabled. The S1D13513 should be in Power Save Mode during this time.

In power save mode, the asynchronous registers (see Section 10.1, “Register Mapping” on page 126) can be accessed.

Note
When Power Save Mode is enabled, synchronous registers and SDRAM memory must not be accessed.

11.2.5 Normal Mode

All functions are available in Normal Mode. However, clocks to modules that are not in use are dynamically stopped. Before enabling Power Save Mode (REG[0470h] bit 0 = 1b) from Normal Mode, confirm that the memory controller is idle (REG[1C02h] bit 7= 1b).

11.2.6 Power-Off

When powering-off the S1D13513, the following sequence must be used.

1. H V_{DD} (HVDD1 ~ HVDD5) Off
2. CORE V_{DD} On, OSC V_{DD}, PLL V_{DD} On

Note

H V_{DD} is 0.8V when CORE V_{DD} is enabled and H V_{DD} is not enabled.

11.3 Power Save Mode Functions

To place the S1D13513 into a power efficient state, perform the following steps.

1. Place the SDRAM into “Self Refresh Mode”.
   a. Set REG[1C06h] bit 14 = 1b.
   b. Set REG[1C08h] bits 1-0 = 11b.
2. Enable Power Save Mode.
   a. Set REG[0470h] = 0001h.

To return the S1D13513 into normal operating mode, perform the following steps.

1. Disable Power Save Mode.
   a. Set REG[0470h] = 0000h.
2. Remove the SDRAM from “Self Refresh Mode”.
   a. Set REG[1C06h] bit 14 = 0b.
   b. Set REG[1C08h] bits 1-0 = 00b.
   c. Set REG[1C0Ah] = 0002h (executes an auto refresh).

12 Data Formats

12.1 Memory Data Formats

The S1D13513 supports 8/16/32 bpp color depths for the Main and PIP1 windows. The selected color depth is controlled by the Main Window Bpp Select bits (REG[0832h] bits 2-0) and the PIP1 Window Bpp Select bits (REG[0832h] bits 6-4), respectively. The image data is stored in memory as shown in the tables below. For 8 bpp, R_0^2 defines the most significant bit of R data for pixel 0.

Table 12-1: 8 Bpp (RGB 3:3:2) Format Data

Address	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0000h	R_1^2	R_1^1	R_1^0	G_1^2	G_1^1	G_1^0	B_1^1	B_1^0	R_0^2	R_0^1	R_0^0	G_0^2	G_0^1	G_0^0	B_0^1	B_0^0
0002h	R_3^2	R_3^1	R_3^0	G_3^2	G_3^1	G_3^0	B_3^1	B_3^0	R_2^2	R_2^1	R_2^0	G_2^2	G_2^1	G_2^0	B_2^1	B_2^0
0004h	R_5^2	R_5^1	R_5^0	G_5^2	G_5^1	G_5^0	B_5^1	B_5^0	R_4^2	R_4^1	R_4^0	G_4^2	G_4^1	G_4^0	B_4^1	B_4^0
0006h	R_7^2	R_7^1	R_7^0	G_7^2	G_7^1	G_7^0	B_7^1	B_7^0	R_6^2	R_6^1	R_6^0	G_6^2	G_6^1	G_6^0	B_6^1	B_6^0

Table 12-2: 16 Bpp (RGB 5:6:5) Format Data

Address	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0000h	R_0^4	R_0^3	R_0^2	R_0^1	R_0^0	G_0^5	G_0^4	G_0^3	G_0^2	G_0^1	G_0^0	B_0^4	B_0^3	B_0^2	B_0^1	B_0^0
0002h	R_1^4	R_1^3	R_1^2	R_1^1	R_1^0	G_1^5	G_1^4	G_1^3	G_1^2	G_1^1	G_1^0	B_1^4	B_1^3	B_1^2	B_1^1	B_1^0
0004h	R_2^4	R_2^3	R_2^2	R_2^1	R_2^0	G_2^5	G_2^4	G_2^3	G_2^2	G_2^1	G_2^0	B_2^4	B_2^3	B_2^2	B_2^1	B_2^0
0006h	R_3^4	R_3^3	R_3^2	R_3^1	R_3^0	G_3^5	G_3^4	G_3^3	G_3^2	G_3^1	G_3^0	B_3^4	B_3^3	B_3^2	B_3^1	B_3^0

Table 12-3: 32 Bpp (RGB 8:8:8) Format Data

Address	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0000h	G_0^7	G_0^6	G_0^5	G_0^4	G_0^3	G_0^2	G_0^1	G_0^0	B_0^7	B_0^6	B_0^5	B_0^4	B_0^3	B_0^2	B_0^1	B_0^0
0002h	—	—	—	—	—	—	—	—	R_0^7	R_0^6	R_0^5	R_0^4	R_0^3	R_0^2	R_0^1	R_0^0
0004h	G_1^7	G_1^6	G_1^5	G_1^4	G_1^3	G_1^2	G_1^1	G_1^0	B_1^7	B_1^6	B_1^5	B_1^4	B_1^3	B_1^2	B_1^1	B_1^0
0006h	—	—	—	—	—	—	—	—	R_1^7	R_1^6	R_1^5	R_1^4	R_1^3	R_1^2	R_1^1	R_1^0

Note

When configured for color passive panels (REG[0800h] bit 14 = 1b and REG[0800h] bits 13-11 = 010b), the maximum number of colors displayed in 32 bpp is 262,144. When the Gamma LUT is not used, only R[7:2], G[7:2], and B[7:2] are used for display. If the Gamma LUT is used, all data bits are used to determine the LUT index, but only Gamma LUT output data bits [7:2] are used for display (LUT output data bits [1:0] are not used).

The maximum number of gray shades displayed in 8 bpp, 16 bpp or 32 bpp is 64. The monochrome pixel data is D[7:0], but only D[7:2] are used for display (D[1:0] are not used). If the Gamma LUT is used, all data bits are used to determine the LUT index, but only LUT output data bits [7:2] are used for display (LUT output data bits [1:0] are not used).

Table 12-4: 8 Bpp Format Data for Monochrome Passive Panels

Address	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0000h	D_1^7	D_1^6	D_1^5	D_1^4	D_1^3	D_1^2	D_1^1	D_1^0	D_0^7	D_0^6	D_0^5	D_0^4	D_0^3	D_0^2	D_0^1	D_0^0
0002h	D_3^7	D_3^6	D_3^5	D_3^4	D_3^3	D_3^2	D_3^1	D_3^0	D_2^7	D_2^6	D_2^5	D_2^4	D_2^3	D_2^2	D_2^1	D_2^0
0004h	D_5^7	D_5^6	D_5^5	D_5^4	D_5^3	D_5^2	D_5^1	D_5^0	D_4^7	D_4^6	D_4^5	D_4^4	D_4^3	D_4^2	D_4^1	D_4^0
0006h	D_7^7	D_7^6	D_7^5	D_7^4	D_7^3	D_7^2	D_7^1	D_7^0	D_6^7	D_6^6	D_6^5	D_6^4	D_6^3	D_6^2	D_6^1	D_6^0

Table 12-5: 16 Bpp Format Data for Monochrome Passive Panels

Address	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0000h	D_0^7	D_0^6	D_0^5	—	—	D_0^4	D_0^3	D_0^2	—	—	—	D_0^1	D_0^0	—	—	—
0002h	D_1^7	D_1^6	D_1^5	—	—	D_1^4	D_1^3	D_1^2	—	—	—	D_1^1	D_1^0	—	—	—
0004h	D_2^7	D_2^6	D_2^5	—	—	D_2^4	D_2^3	D_2^2	—	—	—	D_2^1	D_2^0	—	—	—
0006h	D_3^7	D_3^6	D_3^5	—	—	D_3^4	D_3^3	D_3^2	—	—	—	D_3^1	D_3^0	—	—	—

Table 12-6: 32 Bpp Format Data for Monochrome Passive Panels

Address	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0000h	D_0^4	D_0^3	D_0^2	—	—	—	—	—	D_0^1	D_0^0	—	—	—	—	—	—
0002h	—	—	—	—	—	—	—	—	D_0^7	D_0^6	D_0^5	—	—	—	—	—
0004h	D_1^4	D_1^3	D_1^2	—	—	—	—	—	D_1^1	D_1^0	—	—	—	—	—	—
0006h	—	—	—	—	—	—	—	—	D_1^7	D_1^6	D_1^5	—	—	—	—	—

The PIP2 window supports ARGB, which is used for alpha blending, in addition to the standard 8/16/32 bpp color depths. The PIP2 Window ARGB Format Select bit (REG[0832h] bit 11) determines whether an ARGB or RGB format is specified, while the PIP2 Window Bpp Select bits (REG[0832h] bits 10-8) select the actual format. For ARGB data formats, the image data is stored in memory as shown below.

Table 12-7: 16 Bpp (ARGB 1:5:5:5) Format Data

Address	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0000h	A_0^0	R_0^4	R_0^3	R_0^2	R_0^1	R_0^0	G_0^4	G_0^3	G_0^2	G_0^1	G_0^0	B_0^4	B_0^3	B_0^2	B_0^1	B_0^0
0002h	A_1^0	R_1^4	R_1^3	R_1^2	R_1^1	R_1^0	G_1^4	G_1^3	G_1^2	G_1^1	G_1^0	B_1^4	B_1^3	B_1^2	B_1^1	B_1^0
0004h	A_2^0	R_2^4	R_2^3	R_2^2	R_2^1	R_2^0	G_2^4	G_2^3	G_2^2	G_2^1	G_2^0	B_2^4	B_2^3	B_2^2	B_2^1	B_2^0
0006h	A_3^0	R_3^4	R_3^3	R_3^2	R_3^1	R_3^0	G_3^4	G_3^3	G_3^2	G_3^1	G_3^0	B_3^4	B_3^3	B_3^2	B_3^1	B_3^0

Table 12-8: 16 Bpp (ARGB 4:4:4:4) Format Data

Address	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0000h	A_0^3	A_0^2	A_0^1	A_0^0	R_0^3	R_0^2	R_0^1	R_0^0	G_0^3	G_0^2	G_0^1	G_0^0	B_0^3	B_0^2	B_0^1	B_0^0
0002h	A_1^3	A_1^2	A_1^1	A_1^0	R_1^3	R_1^2	R_1^1	R_1^0	G_1^3	G_1^2	G_1^1	G_1^0	B_1^3	B_1^2	B_1^1	B_1^0
0004h	A_2^3	A_2^2	A_2^1	A_2^0	R_2^3	R_2^2	R_2^1	R_2^0	G_2^3	G_2^2	G_2^1	G_2^0	B_2^3	B_2^2	B_2^1	B_2^0
0006h	A_3^3	A_3^2	A_3^1	A_3^0	R_3^3	R_3^2	R_3^1	R_3^0	G_3^3	G_3^2	G_3^1	G_3^0	B_3^3	B_3^2	B_3^1	B_3^0

Table 12-9: 32 Bpp (ARGB 8:8:8:8) Format Data

Address	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0000h	G_0^7	G_0^6	G_0^5	G_0^4	G_0^3	G_0^2	G_0^1	G_0^0	B_0^7	B_0^6	B_0^5	B_0^4	B_0^3	B_0^2	B_0^1	B_0^0
0002h	A_0^7	A_0^6	A_0^5	A_0^4	A_0^3	A_0^2	A_0^1	A_0^0	R_0^7	R_0^6	R_0^5	R_0^4	R_0^3	R_0^2	R_0^1	R_0^0
0004h	G_1^7	G_1^6	G_1^5	G_1^4	G_1^3	G_1^2	G_1^1	G_1^0	B_1^7	B_1^6	B_1^5	B_1^4	B_1^3	B_1^2	B_1^1	B_1^0
0006h	A_1^7	A_1^6	A_1^5	A_1^4	A_1^3	A_1^2	A_1^1	A_1^0	R_1^7	R_1^6	R_1^5	R_1^4	R_1^3	R_1^2	R_1^1	R_1^0

Note

When configured for Color Passive Panels (REG[0800h] bit 14 = 1b and REG[0800h] bits 13-11 = 010b), the maximum number of colors displayed in 32 bpp is 262,144. When the Gamma LUT is not used, only R[7:2], G[7:2], and B[7:2] are used for display. If the Gamma LUT is used, all data bits are used to determine the LUT index, but only Gamma LUT output data bits [7:2] are used for display (LUT output data bits [1:0] are not used).

The maximum number of gray shades displayed in 8 bpp, 16 bpp, or 32 bpp is 64. The monochrome pixel data is D[7:0], but only D[7:2] are used for display (D[1:0] are not used). If the Gamma LUT is used, all data bits are used to determine the LUT index, but only LUT output data bits [7:2] are used for display (LUT output data bits [1:0] are not used).

Table 12-10: 16 Bpp (ARGB 1:5:5:5) Format Data

Address	**Bit 15**	**Bit 14**	**Bit 13**	**Bit 12**	**Bit 11**	**Bit 10**	**Bit 9**	**Bit 8**	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
0000h	A_0^0	D_0^7	D_0^6	D_0^5	—	—	D_0^4	D_0^3	D_0^2	—	—	D_0^1	D_0^0	—	—	—
0002h	A_1^0	D_1^7	D_1^6	D_1^5	—	—	D_1^4	D_1^3	D_1^2	—	—	D_1^1	D_1^0	—	—	—
0004h	A_2^0	D_2^7	D_2^6	D_2^5	—	—	D_2^4	D_2^3	D_2^2	—	—	D_2^1	D_2^0	—	—	—
0006h	A_3^0	D_3^7	D_3^6	D_3^5	—	—	D_3^4	D_3^3	D_3^2	—	—	D_3^1	D_3^0	—	—	—

Table 12-11: 16 Bpp (ARGB 4:4:4:4) Format Data

Address	**Bit 15**	**Bit 14**	**Bit 13**	**Bit 12**	**Bit 11**	**Bit 10**	**Bit 9**	**Bit 8**	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
0000h	A_0^3	A_0^2	A_0^1	A_0^0	D_0^7	D_0^6	D_0^5	—	D_0^4	D_0^3	D_0^2	—	D_0^1	D_0^0	—	—
0002h	A_1^3	A_1^2	A_1^1	A_1^0	D_1^7	D_1^6	D_1^5	—	D_1^4	D_1^3	D_1^2	—	D_1^1	D_1^0	—	—
0004h	A_2^3	A_2^2	A_2^1	A_2^0	D_2^7	D_2^6	D_2^5	—	D_2^4	D_2^3	D_2^2	—	D_2^1	D_2^0	—	—
0006h	A_3^3	A_3^2	A_3^1	A_3^0	D_3^7	D_3^6	D_3^5	—	D_3^4	D_3^3	D_3^2	—	D_3^1	D_3^0	—	—

Table 12-12: 32 Bpp (ARGB 8:8:8:8) Format Data

Address	**Bit 15**	**Bit 14**	**Bit 13**	**Bit 12**	**Bit 11**	**Bit 10**	**Bit 9**	**Bit 8**	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
0000h	D_0^4	D_0^3	D_0^2	—	—	—	—	—	D_0^1	D_0^0	—	—	—	—	—	—
0002h	A_0^7	A_0^6	A_0^5	A_0^4	A_0^3	A_0^2	A_0^1	A_0^0	D_0^7	D_0^6	D_0^5	—	—	—	—	—
0004h	D_1^4	D_1^3	D_1^2	—	—	—	—	—	D_1^1	D_1^0	—	—	—	—	—	—
0006h	A_1^7	A_1^6	A_1^5	A_1^4	A_1^3	A_1^2	A_1^1	A_1^0	D_1^7	D_1^6	D_1^4	—	—	—	—	—

13 Display Functions

The S1D13513 supports SwivelView and Mirror functions for both the PIP2 window and the View Port (Main+PIP1). These functions are controlled using Display Mode Setting Register 2 (REG[0834h]) and are explained in the following sections.

13.1 SwivelView™

Most computer displays are refreshed in landscape orientation - from left to right and top to bottom. Computer images are stored in the same manner. SwivelView is designed to rotate the displayed image on a LCD by 180° in a counter-clockwise direction.

This rotation is done in hardware and is transparent to the user for all display buffer reads and writes. This is accomplished by rotating the image during display refresh. Therefore, the image is not actually rotated in the display buffer since there is no address translation required during Host CPU reads/writes. By processing the rotation in hardware, SwivelView offers a performance advantage over software rotation of the displayed image.

The SwivelView function can be independently controlled for either the View Port (Main+PIP1) using REG[0834h] bits 1-0, the PIP2 window using REG[0834h] bits 9-8, or both.

13.1.1 0° SwivelView

The following figure shows the relationship between the image stored in the display buffer and the image as displayed on the LCD panel when 0° SwivelView is selected. The image is written to the S1D13513 display buffer in the following sense: A-B-C-D. However, the LCD display is refreshed in the following sense: D-C-B-A.

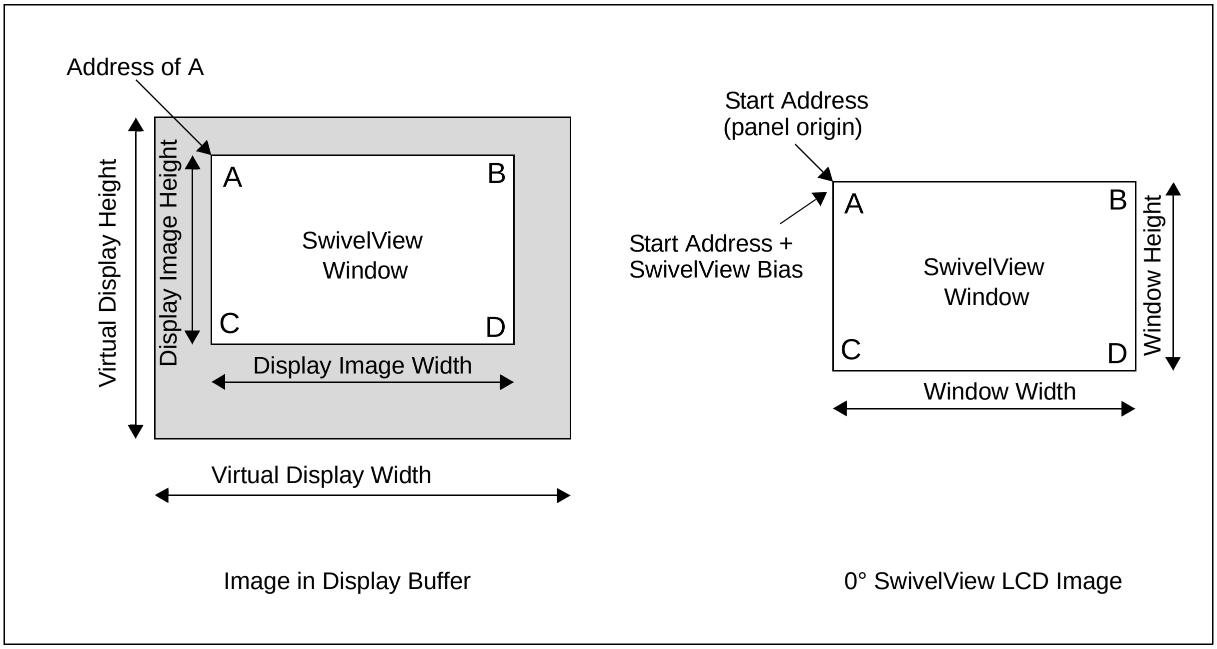


Figure 13-1: Relationship Between Display Buffer Image and LCD Image for 0° SwivelView

Display Start Address

The display refresh circuitry starts at pixel “A”, therefore the Start Address registers must be programmed with the address of pixel “A” and the SwivelView Bias must be set to 0.

SwivelView Bias = 0

Line Address Offset

The Line Address Offset is set as byte counts per 1 line of virtual image.

Line Address Offset = Virtual Image Width x bpp ÷ 8

Memory Address of a Given Pixel

To calculate the address of pixel at any given position for the View Port or PIP2 window, use the following formula.

Memory Address (X,Y) = [(X - 1) + (Y - 1) x virtual panel height] x bpp ÷ 8

13.1.2 180° SwivelView

The following figure shows the relationship between the image stored in the display buffer and the image as displayed on the LCD panel when 180° SwivelView is enabled. The image is written to the S1D13513 display buffer in the following sense: A-B-C-D. However, the LCD display is refreshed in the following sense: D-C-B-A.

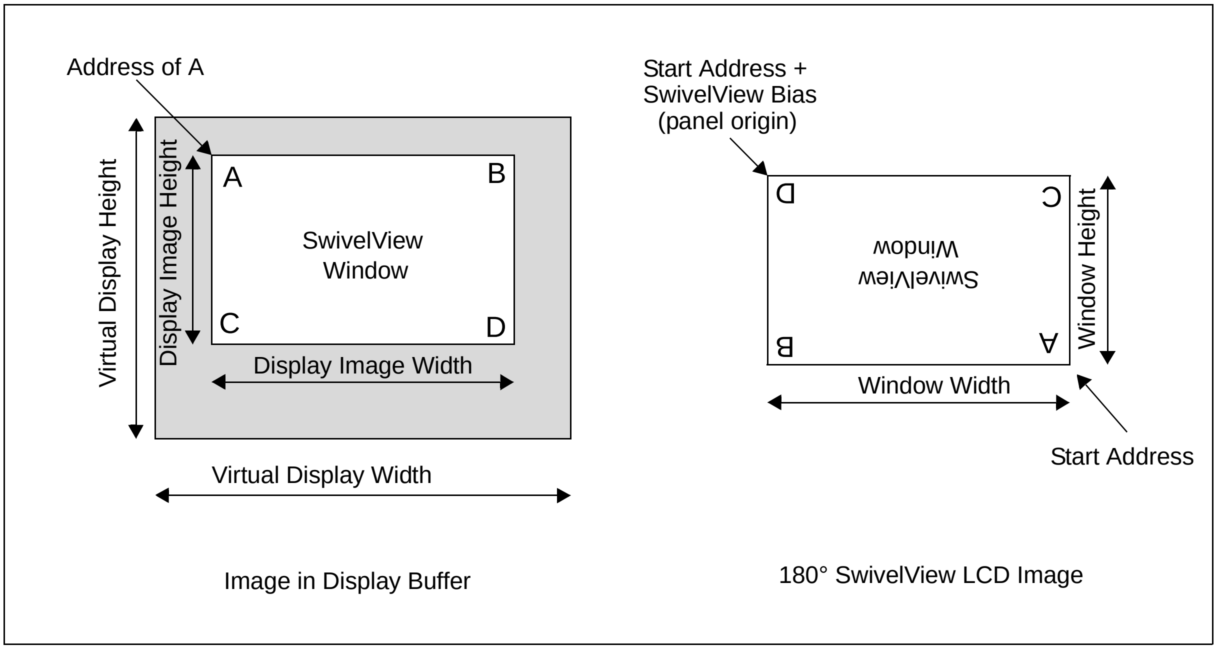


Figure 13-2: Relationship Between Display Buffer Image and LCD Image for 180° SwivelView

Display Start Address

The display refresh circuitry starts at pixel “D”, therefore the Start Address registers must be programmed with the address of pixel “A” and the SwivelView Bias should be programmed as follows.

SwivelView Bias = Line Address Offset x Window Height - (bpp ÷ 8)

Line Address Offset

The Line Address Offset is set as byte counts per 1 line of virtual image.

Line Address Offset = Virtual Image Width x bpp ÷ 8

Memory Address of a Given Pixel

To calculate the address of pixel at any given position for the View Port or PIP2 window, use the following formula.

Memory Address (X,Y) = [(X - 1) + (Y - 1) x virtual panel height] x bpp ÷ 8

13.2 Mirror Display

Most computer displays are refreshed from left to right and top to bottom. Computer images are stored in the same manner. Mirror Display is designed to refresh the display from right to left, thus “mirroring” the display. Mirror Display is performed by hardware and no changes in the storage of display data in the display buffer are required. By mirroring the image in hardware, Mirror Display offers a performance advantage over software mirroring of the same image.

Mirror Display can be independently enabled for either the View Port (Main+PIP1) using REG[0834h] bit 2, the PIP2 window using REG[0834h] bit 10, or both.

13.2.1 Mirror Display for 0° SwivelView

The following figure shows the relationship between the image stored in the display buffer and the image as displayed on the LCD panel when no rotation is selected (0° SwivelView). The image is written to the S1D13513 display buffer in the following sense: A-B-C-D. However, the LCD display is refreshed in the following sense: B-A-D-C.

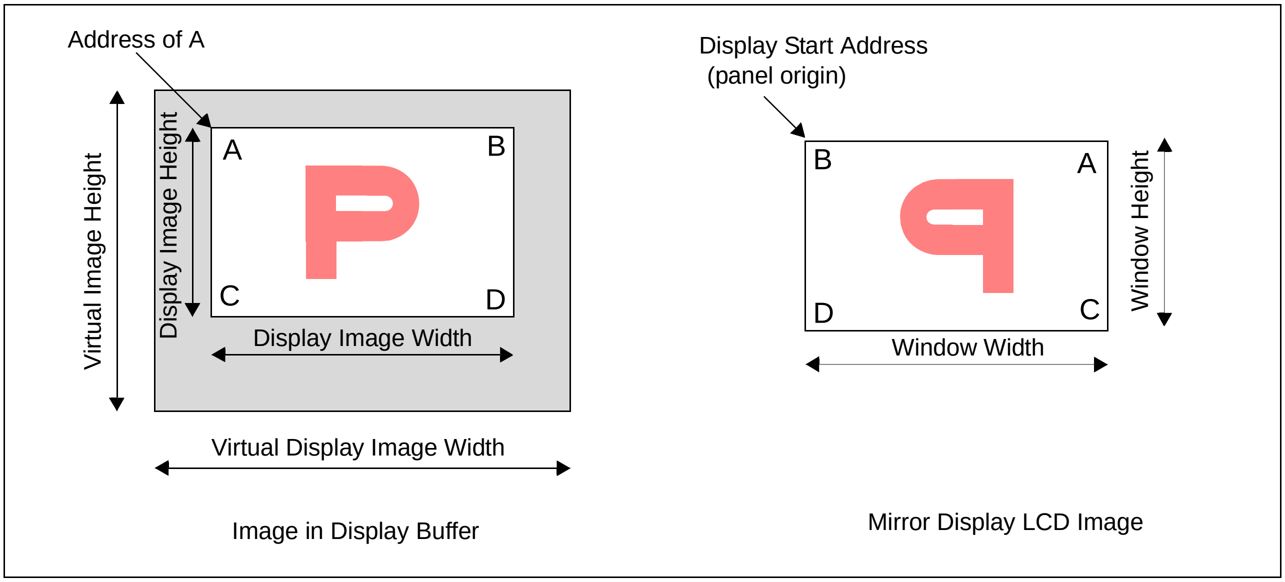


Figure 13-3: Relationship Between Display Buffer Image and LCD Image for Mirror Display (0° SwivelView)

Display Start Address

The display refresh circuitry starts at pixel “B”, therefore the Display Start Address registers must be programmed with the address of pixel “B”.

Display Start Address= Address of A + Line Address Offset - (bpp ÷ 8)

Line Address Offset

The Line Address Offset is set to the number of bytes per line of virtual image.

Line Address Offset = Virtual Image Width x bpp ÷ 8

13.2.2 Mirror Display Combined with SwivelView Modes

When both Mirror Display and SwivelView are enabled, the image is rotated by the SwivelView function after the Mirror Display function takes place. The Display Start Address must be set to the left upper pixel of display image.

Mirror Display with 180° SwivelView

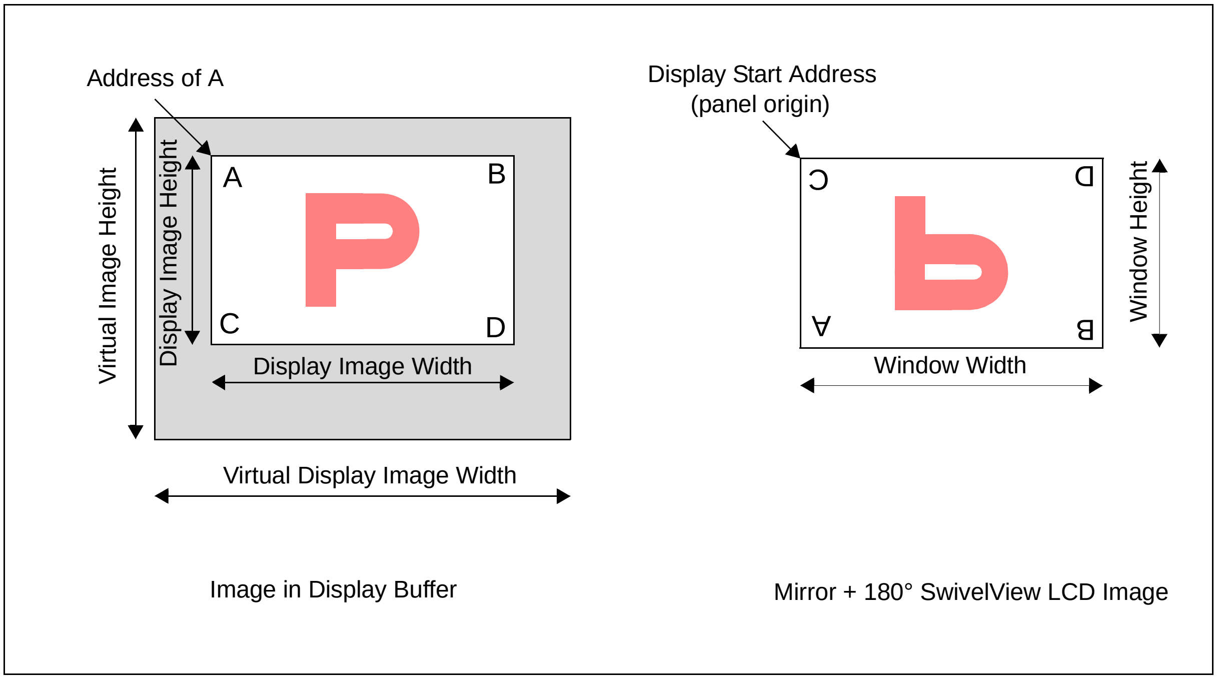


Figure 13-4: Mirror Display with 180° SwivelView Display

13.3 Gamma Correction

The S1D13513 performs gamma correction using two Look-up Tables (LUTs) which are referred to as Bank A and Bank B. Each LUT has 256 (8-bit) entries for each RGB color component. The Bank used for gamma correction is selected using REG[083Eh] bits 5-4.

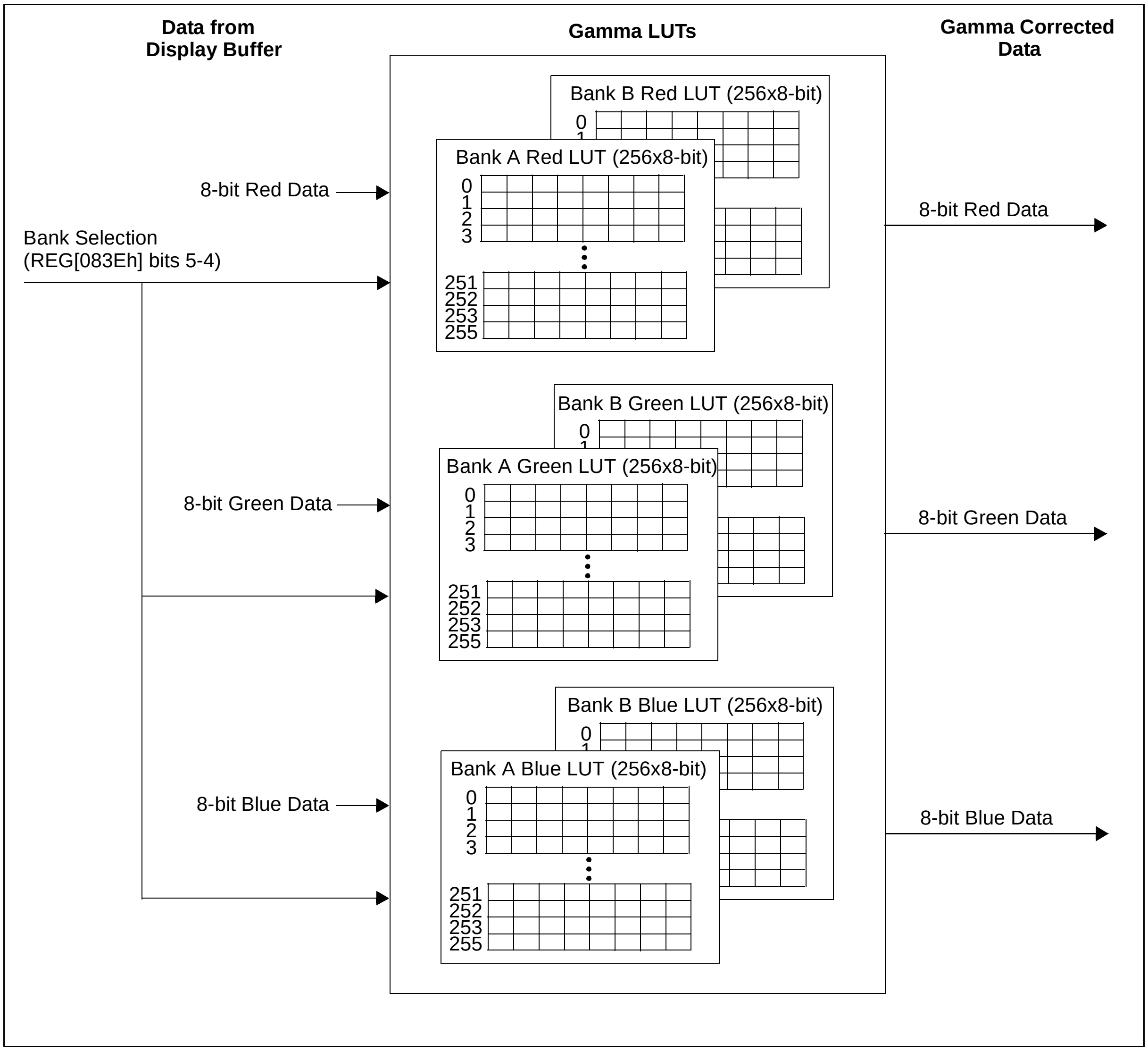


Figure 13-5: Gamma LUT Architecture for Color Modes

Note

In 16 bpp and 8 bpp, the extra bits required for the LUT index are created according to the setting of the Input Data Extra Bit Expansion Enable bit, REG[083Eh] bit 3.

In Monochrome mode (REG[0800h] bit 14 = 1b and REG[0800h] bits 13-11 = 000b), only the green Look-up Tables are used. As in color modes, the S1D13513 performs gamma correction using two Look-up Tables (LUTs) which are referred to as Bank A and Bank B. Each LUT has 256 (8-bit) entries for each pixel. The Bank used for gamma correction is selected using REG[083Eh] bits 5-4.

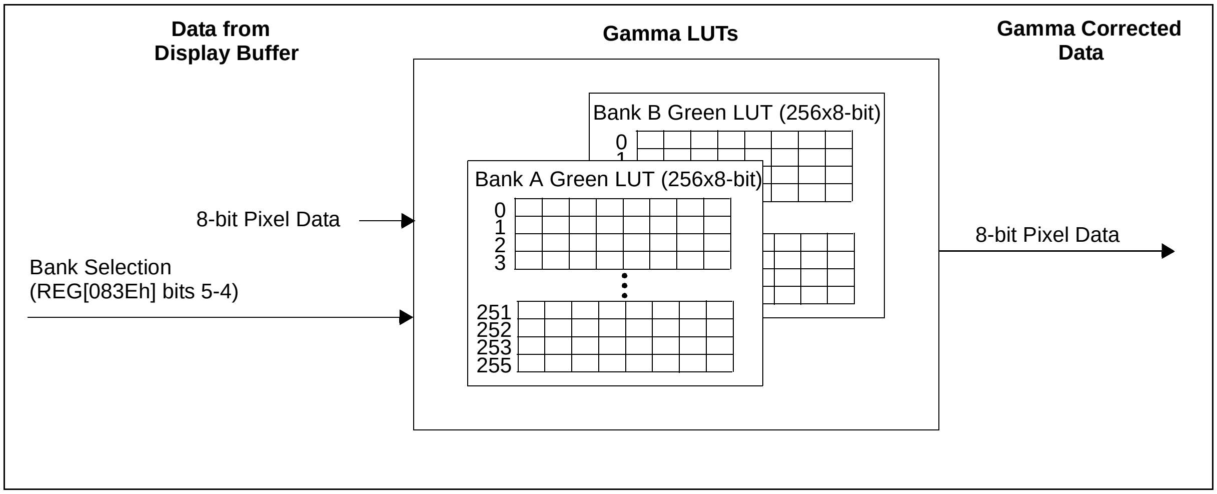


Figure 13-6: Gamma LUT Architecture for Monochrome Modes

13.3.1 Gamma LUT Bank Selection

There are four bank selection modes (see REG[083Eh] bits 5-4) which determine the windows (Main, PIP1, or PIP2) that are gamma corrected using Bank A or Bank B data. Two bank selection modes allow Gamma Correction using one Bank while the other Bank is being programmed. One mode allows different windows to be gamma corrected based on different banks, and one mode allows for both banks to be programmed together.

The following bank selection modes are available.

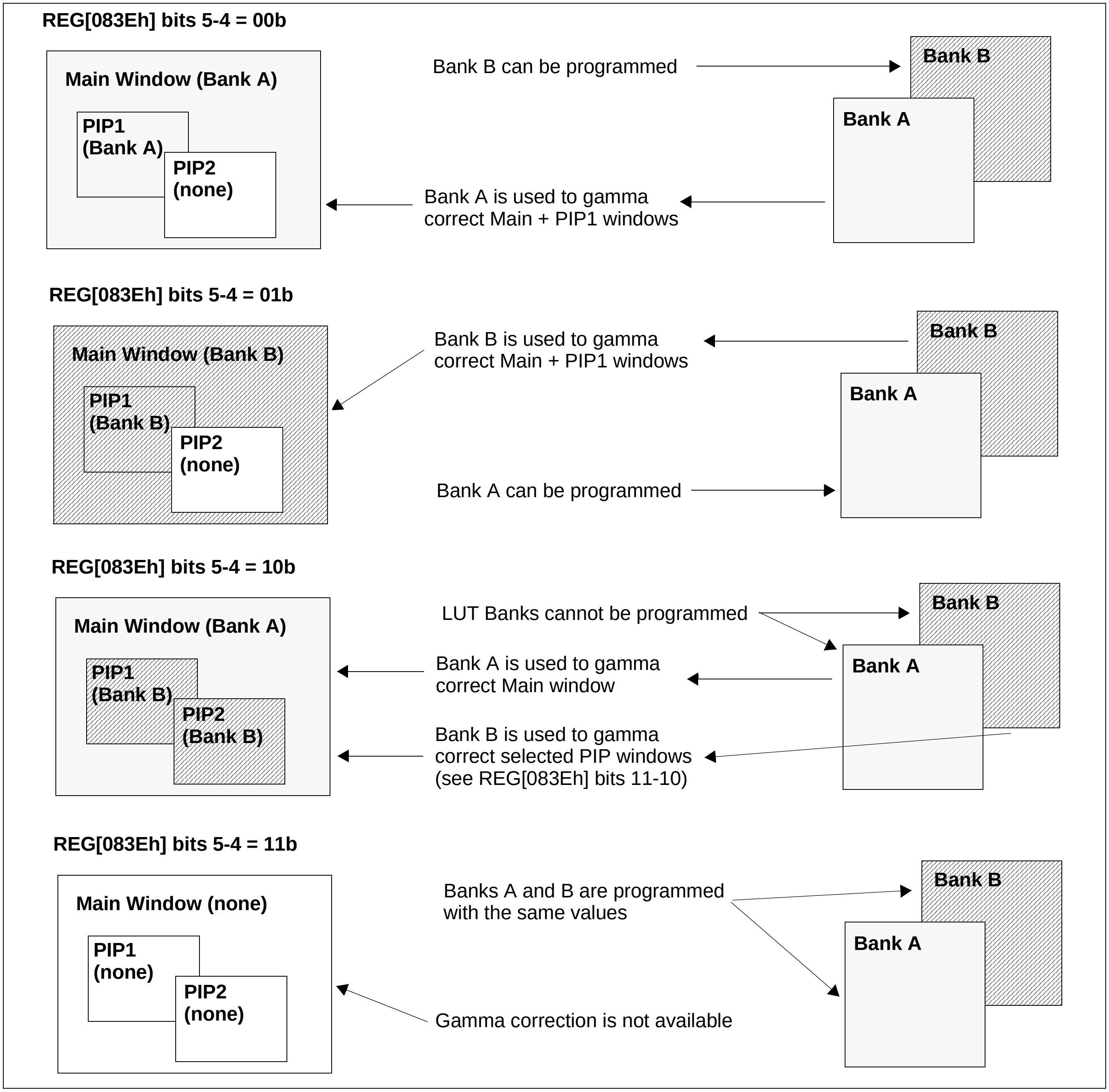

REG[083Eh] bits 5-4 = 00b
Main Window (Bank A)
PIP1 (Bank A)
PIP2 (none)
Bank B can be programmed
Bank A is used to gamma correct Main + PIP1 windows
Bank B
Bank A
REG[083Eh] bits 5-4 = 01b
Main Window (Bank B)
PIP1 (Bank B)
PIP2 (none)
Bank B is used to gamma correct Main + PIP1 windows
Bank A can be programmed
Bank B
Bank A
REG[083Eh] bits 5-4 = 10b
Main Window (Bank A)
PIP1 (Bank B)
PIP2 (Bank B)
LUT Banks cannot be programmed
Bank A is used to gamma correct Main window
Bank B is used to gamma correct selected PIP windows (see REG[083Eh] bits 11-10)
Bank B
Bank A
REG[083Eh] bits 5-4 = 11b
Main Window (none)
PIP1 (none)
PIP2 (none)
Banks A and B are programmed with the same values
Gamma correction is not available
Bank B
Bank A

Figure 13-7: Gamma LUT Bank Selection

13.3.2 Programming the Gamma LUTs

The following procedure should be used to program the Gamma LUTs.

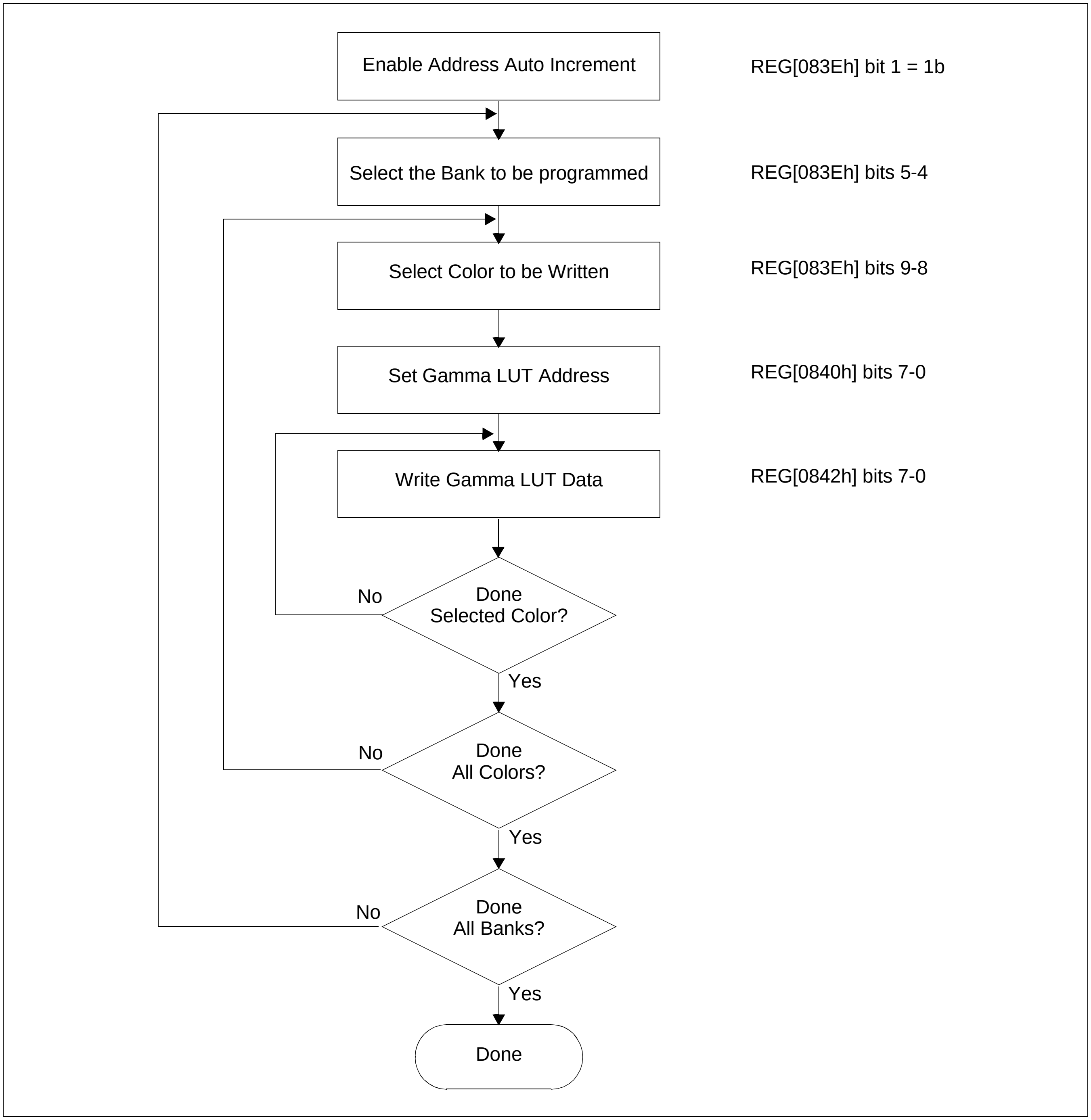


Figure 13-8: Gamma LUT Programming Flowchart

13.4 Pseudo Color Mode

The S1D13513 supports Pseudo Color Mode which is used to minimize the change in image color quality when the color depth of the image is lowered for output to a panel. This is necessary for panels with a smaller data bus width than the display output size. For example if the S1D13513 is configured for a 16-bit TFT panel, the internal display image data is RGB 8:8:8 (24-bit) but must be converted to RGB 5:6:5 (16-bit) for output to the panel.

When Pseudo Color Mode is disabled (REG[0844h] bits 2-0 = 000b), each color component (RGB) is rounded down to the data bus width of the panel. When enabled, the following are selectable.

- 2x2 Matrix Dither - REG[0844h] bits 2-0 = 001b
- FRM - REG[0844h] bits 2-0 = 010b
- Error Diffusion - REG[0844h] bits 2-0 = 100b

To determine the optimum Pseudo Color Mode, it may be necessary to perform a visual evaluation for each specific implementation.

Note

When the S1D13513 is configured for a passive panel (REG[0800h] bit 14 = 1b), the Pseudo Color Mode function operates as dithering.

When dithering is disabled (REG[0844h] = 000b), each color component can display up to 16 shades. This means a color passive panel will display a maximum of 4096 colors and a monochrome panel will display a maximum of 16 gray shades. There is only one type of dithering operation and it is enabled by setting REG[0844h] to a non-zero value (01h, 02h, or 04h).

When the dithering is enabled, each color component can display up to 64 shades. This means a color passive panel will display a maximum of 262,144 colors and a monochrome panel will display a maximum of 64 gray shades.

14 Resizers

The resizer module allows the S1D13513 to resize YUV camera input data and YUV data from the SDRAM. There are two resizers: the View Resizer for viewing image data and the Capture Resizer for capturing image data. It is possible to use both resizers simultaneously. Resizers “resize” image data using trimming and scaling functions.

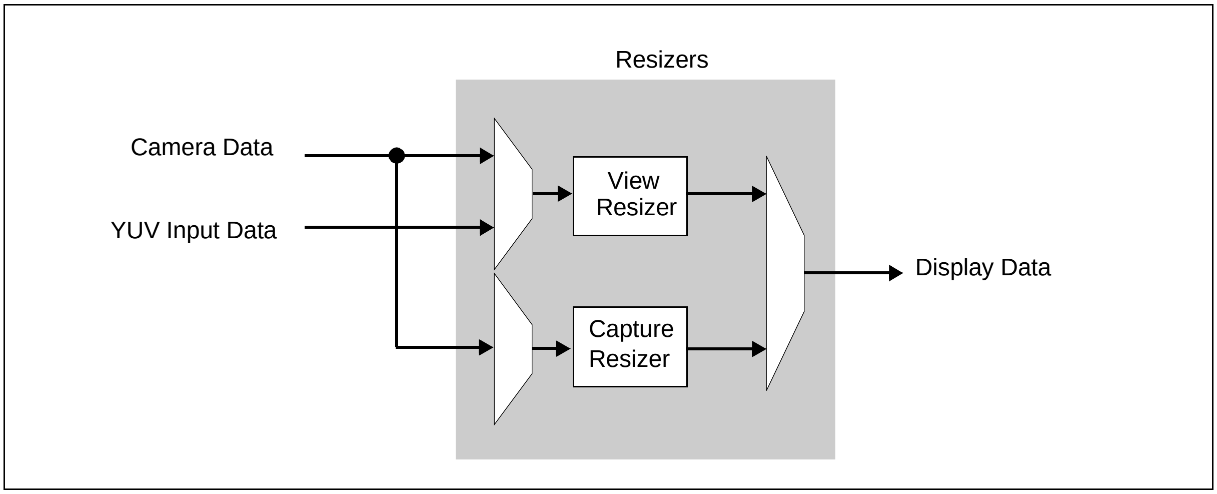


Figure 14-1: Resizer Block Diagram

14.1 Resizer Descriptions

The View Resizer is used for resizing YUV data that will be viewed on the LCD display. It can receive YUV input from the camera interface or YUV data from the SDRAM.

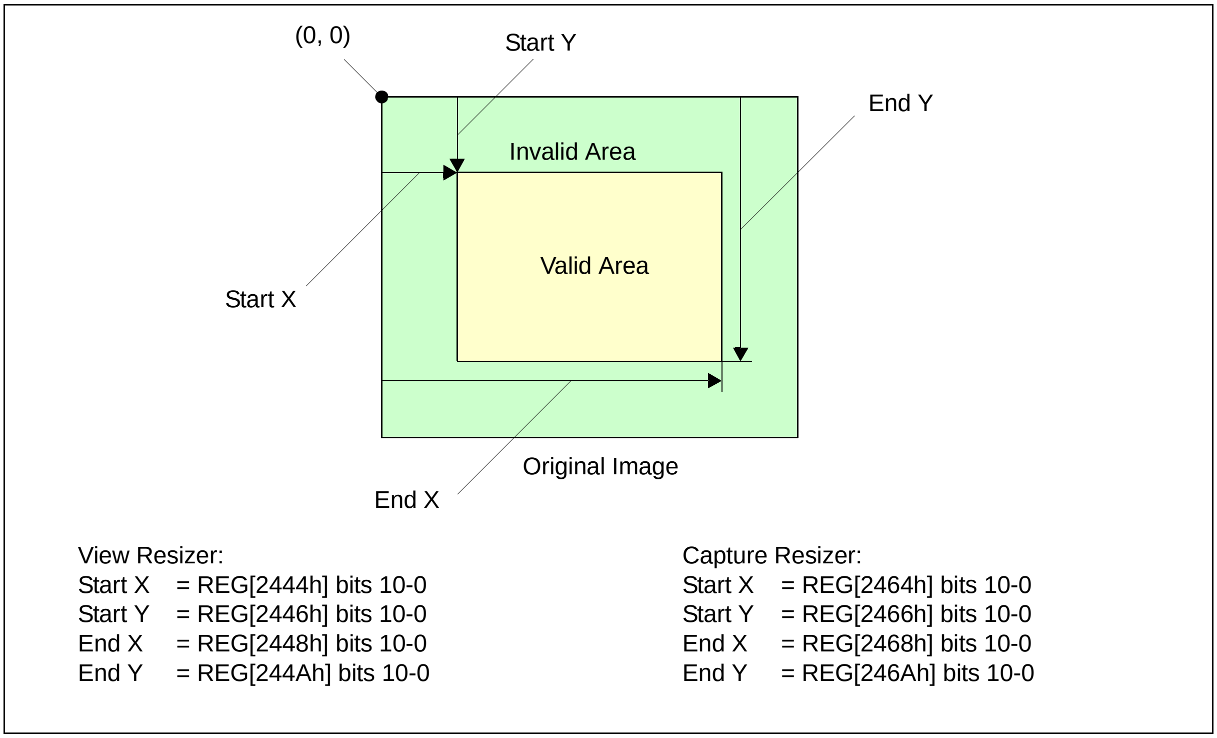
The Capture Resizer is used for resizing YUV data that will be captured and placed in the SDRAM memory. The Capture Resizer receives input only from the camera interface.

Table 14-1: Resizer Selection

Usage	View Resizer	Capture Resizer
Camera Image Display	available	available
YUV Image Data from SDRAM	available	not available

14.2 Trimming Function

The trimming function is similar to cropping an image and "trims" the unwanted portion of the image. The trimming is controlled using the Resizer X/Y Start/End Position registers (REG[2444h] ~ REG[244Ah] or REG[2464h] ~ REG[246Ah]). The Start and End addresses programmed in these registers are limited by the size of the actual camera image and must not be set to a value greater than this actual size. The Start and End Position registers are set in 1 pixel increments.



View Resizer:

Start X	= REG[2444h] bits 10-0
Start Y	= REG[2446h] bits 10-0
End X	= REG[2448h] bits 10-0
End Y	= REG[244Ah] bits 10-0

Capture Resizer:

Start X	= REG[2464h] bits 10-0
Start Y	= REG[2466h] bits 10-0
End X	= REG[2468h] bits 10-0
End Y	= REG[246Ah] bits 10-0

Figure 14-2: Trimming Function

14.3 Scaling Function

The scaling function takes place after the trimming stage and it specifies the desired compression ratio to be applied to the image. The scaling function is independent in the horizontal and vertical directions and scaling rates from 128/128 through 1/128 are available. For 1/2, 1/4, 1/8, 1/16, 1/32, 1/64 and 1/128 scaling, only the horizontal direction can be averaged.

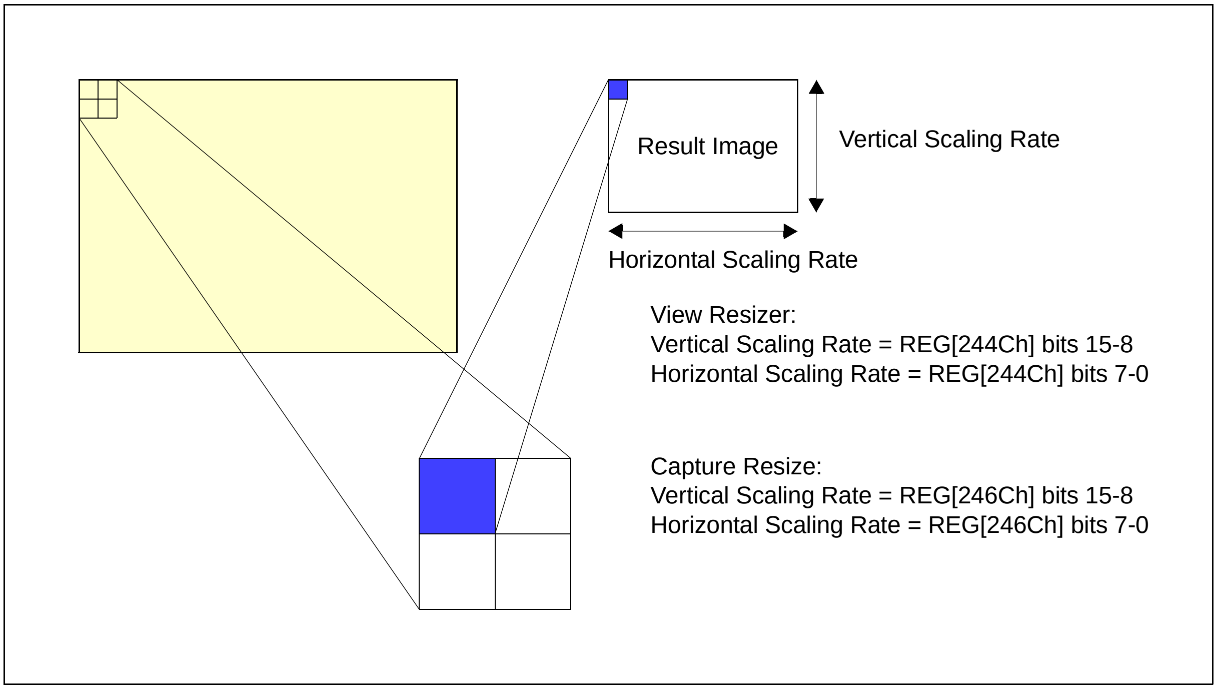

Result Image
Vertical Scaling Rate
Horizontal Scaling Rate
View Resizer:
Vertical Scaling Rate = REG[244Ch] bits 15-8
Horizontal Scaling Rate = REG[244Ch] bits 7-0
Capture Resize:
Vertical Scaling Rate = REG[246Ch] bits 15-8
Horizontal Scaling Rate = REG[246Ch] bits 7-0

Figure 14-3: Scaling Function

14.3.1 Odd Number Scaling

For odd number scaling, one pixel is extracted from the center of the block. Both the horizontal and vertical directions use the reduction method.

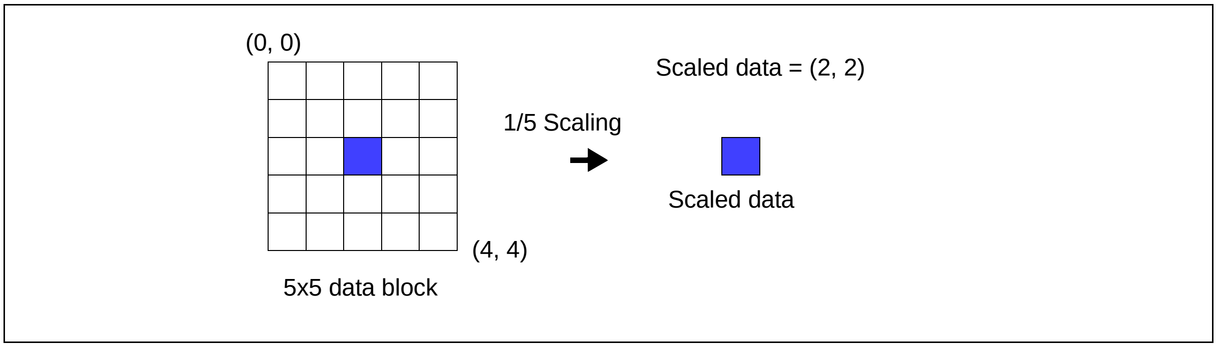


Figure 14-4: Odd number Scaling (Example: 1/5 scaling)

14.3.2 Even Number Scaling

For even number scaling, one pixel is extracted from the center of the block (as shown). Both the horizontal and vertical directions use the reduction method.

Note

For scaling ratios of 1/2, 1/4, 1/8, 1/16, 1/32 1/64 and 1/128 an horizontal average method can be used (see Section 14.3.3, “Averaging Method” on page 396).

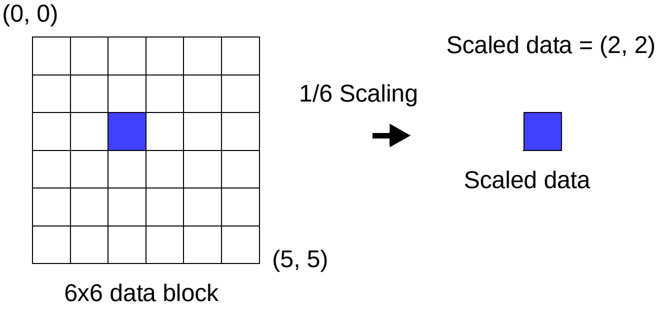


Figure 14-5: Even number Scaling (Example: 1/6 scaling)

14.3.3 Averaging Method

For scaling ratios of 1/2, 1/4, 1/8, 1/16, 1/32, 1/64 and 1/128 one pixel is extracted from the center of the block (as shown). However, the horizontal direction is determined using an average function. The vertical direction uses the reduction method.

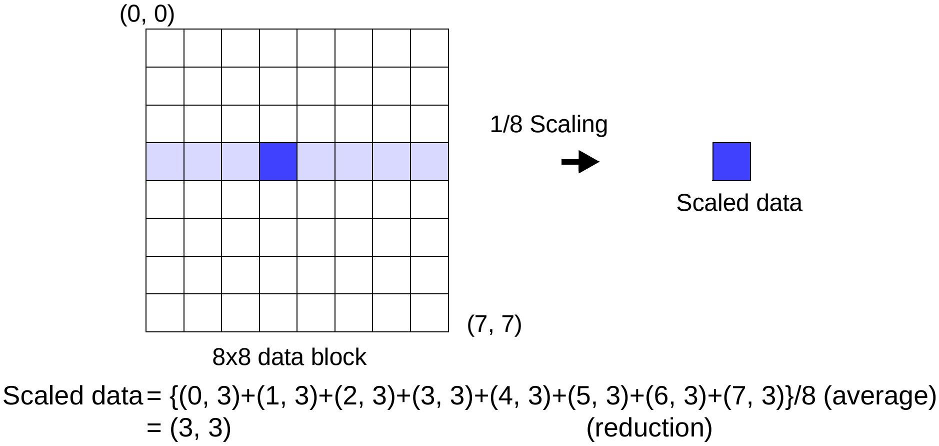

(0, 0)
1/8 Scaling
Scaled data
(7, 7)
8x8 data block
Scaled data = {(0, 3)+(1, 3)+(2, 3)+(3, 3)+(4, 3)+(5, 3)+(6, 3)+(7, 3)}/8 (average)
= (3, 3) (reduction)

Figure 14-6: Average Method (Example: 1/8 scaling)

14.3.4 Calculating the Number of Pixels After Scaling

Definition: (Unit: Pixel)

1. Size after trimming: horizontal is “A” and vertical is “B”
2. Size after scaling: horizontal is “a” and vertical is “b”
3. Reduction rate: horizontal is “X/128” and Vertical is “Y/128”

a = Integer value of (A x X/128)
b = Integer value of (B x Y/128)

Note

The a and b calculation is the same for YUV format. However, the units of a and b after resized is as follows:
YUV 4:2:2 - a: two pixels, b: one pixel
YUV 4:2:0 - a: two pixels, b: two pixels

15 2D BitBLT Engine

15.1 Overview

The purpose of the BitBLT Engine is to off load the work of the CPU for moving pixel data to and from the CPU and display memory and also for moving pixel data from one location to another in display memory.

There are 5 BitBLTs (Bit Block Transfer) which are used to move pixel data from one location to another.

- **Write BitBLT:** Move pixel data from CPU to Display Memory
- **Read BitBLT:** Move pixel data from Display Memory to CPU
- **Move BitBLT**: Move pixel data from one location in Display Memory to another
- **Pattern Fill BitBLT**: Move a Pixel Pattern in Display Memory and duplicate several times to produce a larger image
- **Solid Fill BitBLT**: fills a location in display memory with a data pattern

The BitBLT Engine also can perform several Data Functions in combination with some of the BitBLT functions on the pixel data.

- **Color Expansion**: Expand a Font bit pattern to a higher color depth.
- **ROP**: Perform a Boolean function on the pixel data
- **Transparency**: Only write pixel data of which the color does not match the Transparent Color.

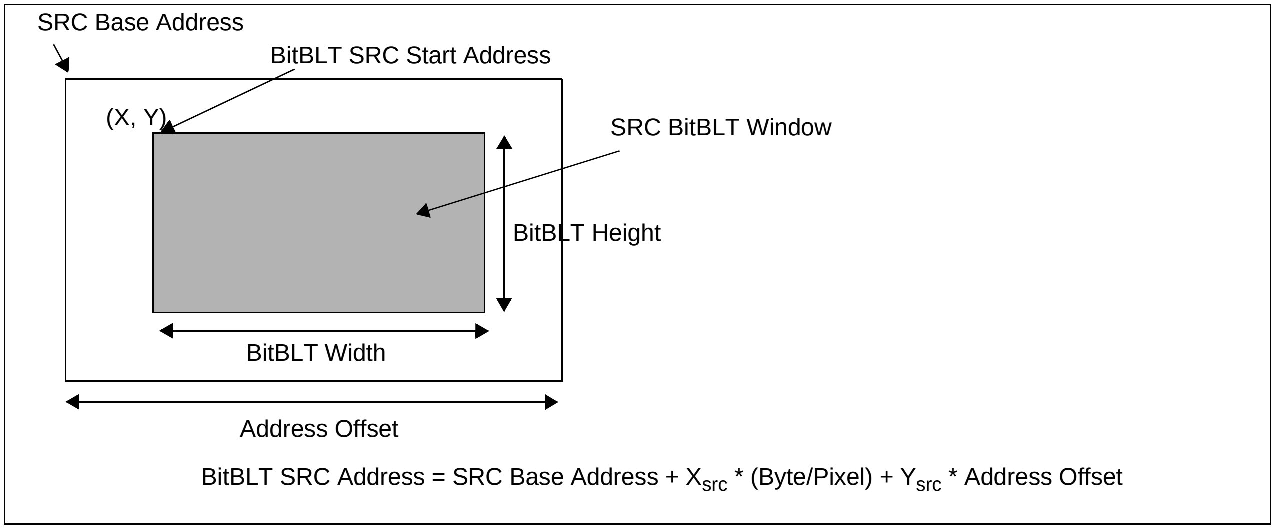
The BitBLT Engine can work with several color depths for the pixel data of 8 bpp, 16 bpp, or 24 bpp (32 bpp).

The following table shows which data function can be used with the different BitBLTs. For details on using these operations and functions, refer to the REG[1808h] bits 2-0 bit description.

Table 15-1 : BitBLT Operation Summary

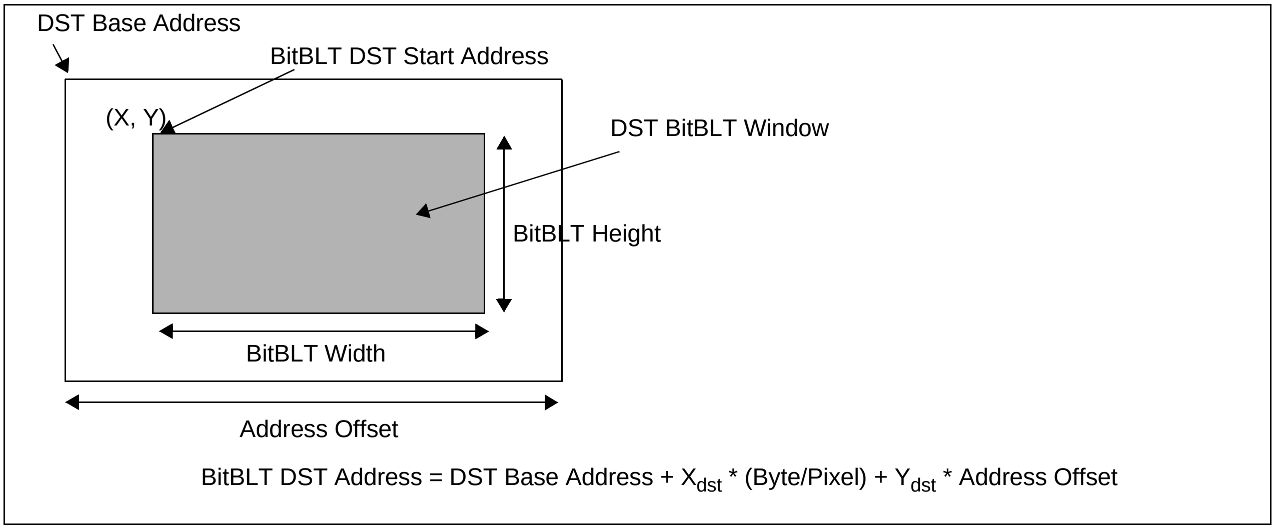
REG[1808h] bits 2-0	BitBLT Operation	Alpha Blending	Clipping	ROP	Transparent	Alpha Combine	Color Expansion	Reverse Direction
000b	Write	—	—	—	√	√	√	—
001b	Read	—	—	—	—	—	—	—
010b	Move	√	√	√	√	—	√	√
011b	Solid Fill	—	—	—	—	—	—	—
100b	Pattern Fill with transparency	—	—	—	—	—	—	—
101b - 111b	Reserved	—	—	—	—	—	—	—

15.2 BitBLT Terms and Definition



BitBLT SRC Address = SRC Base Address + X_{src} * (Byte/Pixel) + Y_{src} * Address Offset

Figure 15-1: BitBLT Source Area Definition



BitBLT DST Address = DST Base Address + X_{dst} * (Byte/Pixel) + Y_{dst} * Address Offset

Figure 15-2: BitBLT Destination Area Definition

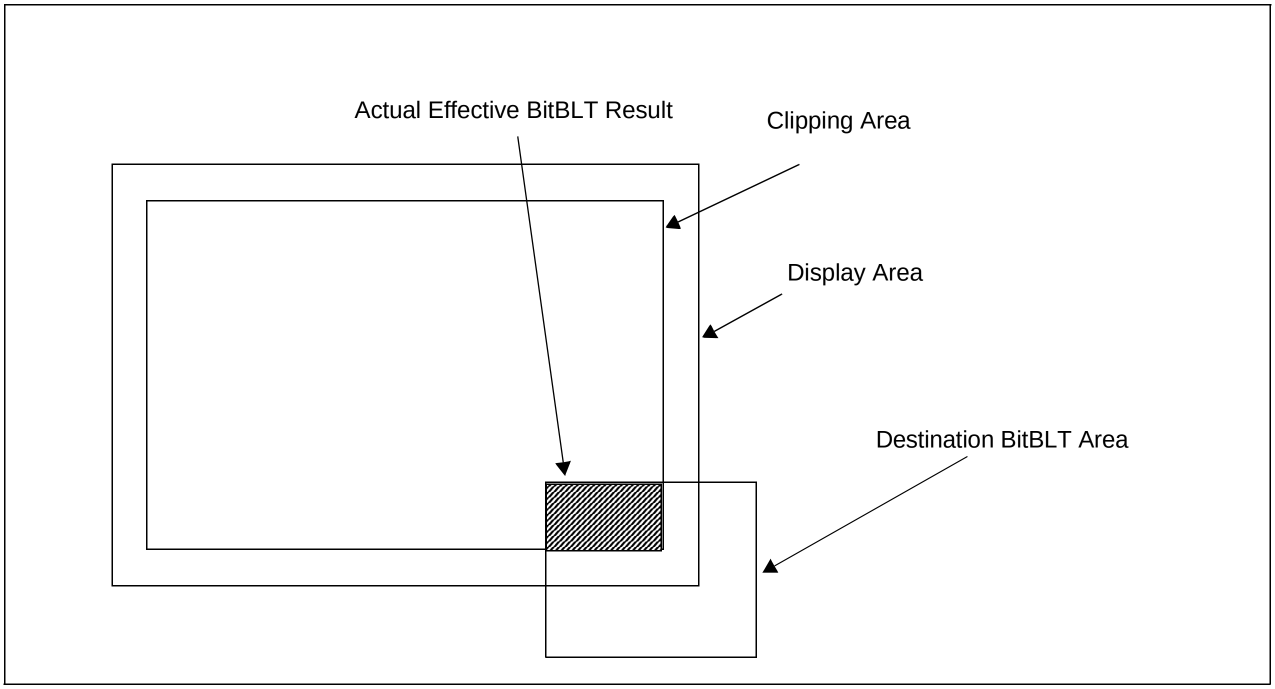
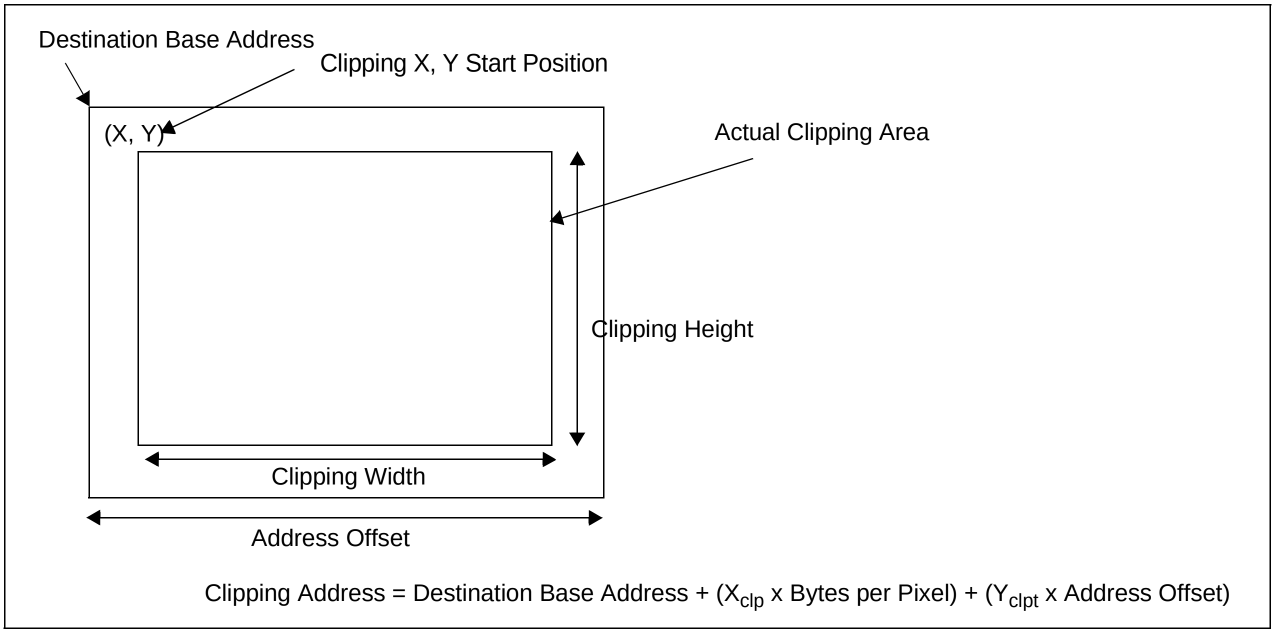


Figure 15-3: BitBLT Clipping Image



Clipping Address = Destination Base Address + (X_{clp} x Bytes per Pixel) + (Y_{clpt} x Address Offset)

Figure 15-4: Clipping Area Definition

Address Offset Width of the display (i.e. Main Window width or PIP^+ Window width) in bytes. The source and destination share the memory address offsets.

SRC Start Address Top left corner of the BitBLT window source specified in bytes. See the diagram for address the calculation.

DST Start Address Top left corner of the BitBLT window destination specified in bytes. See the diagram for address the calculation.

X_{src} Position X- Location value from top left corner for SRC BitBLT window.

Y_{src} Position Y- Location value from top left corner for SRC BitBLT window.

X_{dst} Position X- Location value from top left corner for DST BitBLT window.

Y_{dst} Position Y- Location value from top left corner for DST BitBLT window.

BitBLT Width Width of the BitBLT in pixels. This parameter is shared between Source and Destination.

BitBLT Height Height of the BitBLT in pixels.

BitBLT Window The area of the display memory to work with.

Clipping Area The area that BitBLT result only effectively transfer. For outside clipping area, BitBLT operation will not be done.

Clipping Start Address Top left corner of the Clipping area specified in bytes. See the diagram for address the calculation.

X_{clp} Position X- Location value from top left corner for Clipping area window.

Y_{clp} Position Y- Location value from top left corner for Clipping area window.

For each BitBLT there is a source of data and a destination for the result data. The source is the location where the data is read from. The destination is where the data is written to.

16 Sprite Engine

The S1D13513 is designed with a Sprite Engine to enhance the performance of handheld based games and other applications requiring independent object based graphics. The Sprite Engine allows these objects to be defined as “sprites” which can be easily moved over another image without modifying the background image.

The S1D13513 Sprite Engine features the following.

- Conformity with the Mobile Information Device Profile v2.0 Sprites Class for Java 2 Micro Edition.
- Support for up to 16 individual sprites to be simultaneously displayed. The main image is defined as Sprite #0.
- Sprite #0 can support a virtual image larger than the actual display size.
- Programmable Sprite Size Register - each sprite can vary in size and is only limited by the amount of available memory in SDRAM).
- Individual Sprite X,Y location register - location can be negative on all edges of display allowing the sprite to gradually move off any side of the display.
- Individual Sprite Z-Order (each sprite has an associated z-order which determines which sprite is visible over another sprite when their locations overlap).
- Alpha blending support for all ARGB format sprites
- Sprite image data can use 4 different color formats: RGB 5:6:5, ARGB 1:5:5:5, ARGB 4:4:4:4, or ARGB 8:8:8:8.
- Sprite Rotation / Mirror functions
  - sprite rotation is independent of the main display orientation
  - programmable rotation reference point (by X/Y offsets from the upper left corner of the sprite, both can be positive or negative)
  - Optional Arbitrary Angle Rotation
- Sprite Frame Sequencing supports up to 16 frames for each sprite. Frame sequence length can be from 1 to 16 and be in any specified order. A single frame can be repeated in different locations in the sequence more than once. Sequence can be advanced manually by register bits, or by every programmable number of VSyncs.
- Sprite images are stored in SDRAM.
- Frame rate of greater than 15 fps.
- Any combination of rotation and mirroring can be generated from only 0 degree and 90 degree versions of stored sprite images.
- Supports both single and double buffering.

16.1 Sprite Data Path

All individual sprites are stored in SDRAM. When required, sprite data is read from the SDRAM and synthesized back to SDRAM for display on the panel. Double buffering can be enabled to reduce tearing and allow faster frame rates.

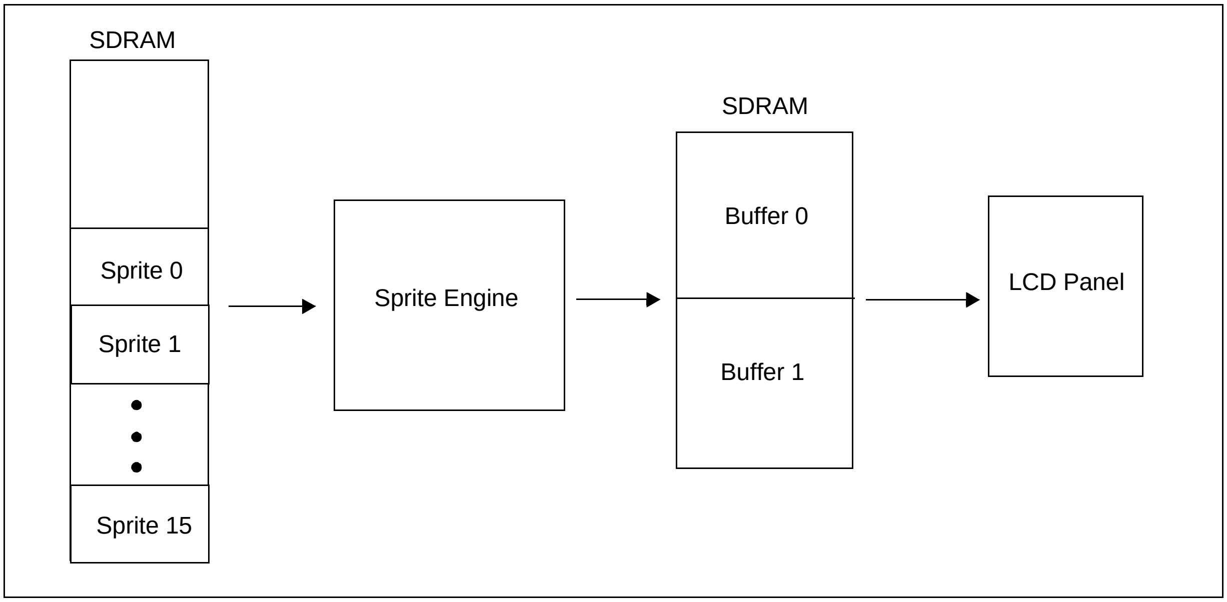


Figure 16-1: Sprite Data Path

16.2 16 Sprite Support with Z-ordering Transparency

Each sprite has an associated z-order which is used to determine which part of the sprite is displayed when the sprite overlaps the main image or other sprites.

Note

Sprite #0 is used as the background and must not include any transparent pixels.

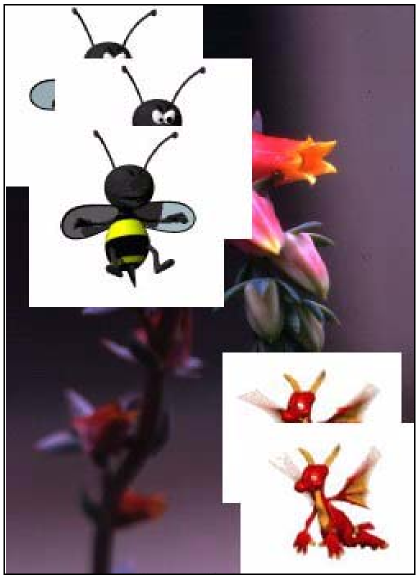


Figure 16-2: Z-order Example

When RGB 5:6:5 format is selected, one programmable transparency color may be associated with it. Transparency allows an irregular shaped image to be displayed over the background.

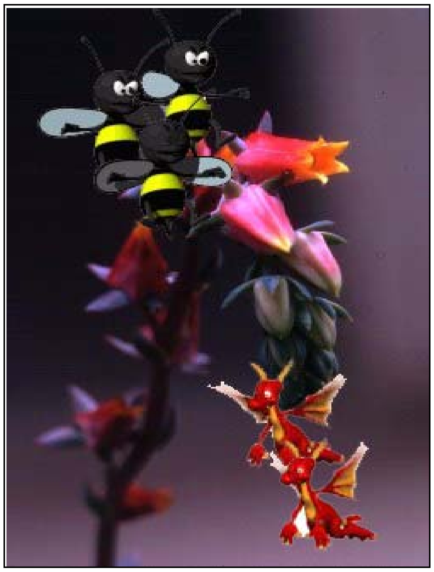


Figure 16-3: Z-order with Transparency Example

16.3 16 Sprite Support with Z-ordering Alpha-Blending

The S1D13513 Sprite Engine supports Alpha-Blending which provides further visual enhancement for games and similar applications. Alpha-blending is used in computer graphics to create the effect of transparency. This technique is useful for graphics that feature glass or liquid objects and is done by combining a translucent foreground with a background color to create a blend. It can also be used for animation, where one image gradually fades into another image.

Note

Sprite #0 is used as the background and must not include any pixels with an alpha value other than FFh (opaque).

The S1D13513 Sprite Engine supports alpha-blending using an 8-bit alpha value (ARGB 8:8:8:8).

- ARGB 1:5:5:5 - one Alpha bit points to 2 programmable indexed 8-bit alpha values
- ARGB 4:4:4:4 - the four bits represent the actual alpha value

The following equation describes the alpha blending technique used.

$$[r, g, b]_{\text{blended}} = \alpha[r, g, b]_{\text{foreground}} + (1 - \alpha)[r, g, b]_{\text{background}}$$

Where:

[r,g,b] are the red, green, and blue color channels
α is the weighting factor

The weighting factor value can be from 0 to 1 (represented as 0 to 255 for S1D13513 Sprite Engine). When set to 0, the foreground is completely transparent. When it is set to 1, the background is completely transparent. All values between specify a mixture of the foreground and the background.

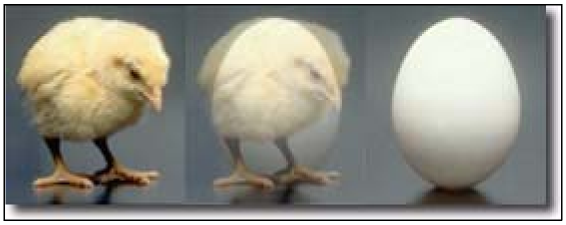


Figure 16-4: Alpha Blending with Alpha Value of 0, 0.5 and 1

The S1D13513 Sprite Engine allows up to 16 sprites to be alpha blended together. Z-ordering determines which sprites are displayed in the foreground and background for each alpha-blending operation.

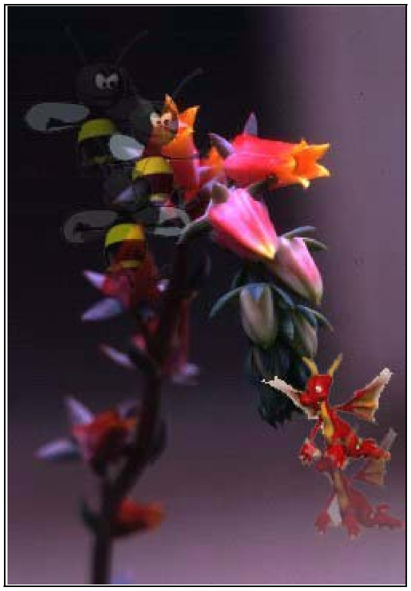


Figure 16-5: Z-order with Alpha-Blending

16.4 16 Sequences for Each Sprite

When enabled, a sequence of frames can be displayed one after the other. The trigger for the sequence can be configured as a register write (i.e. Next Frame) or a programmable number of VSYNCs for each frame. Each sprite can has individual sequence images, sequence length, VSync number, and sequence enable bit. Every sprite also has a frame sequence register, which contains the indexes to each of the 16 frames in the sprite. Various animation effects can be achieved by programming the sequence length and VSync number. The same image can also be repeated within the 16 frames of the sprite.

Sequence Number	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Index Assignment	0	1	2	3	4	4	4	5	5	6	7	8	9	9	0	1

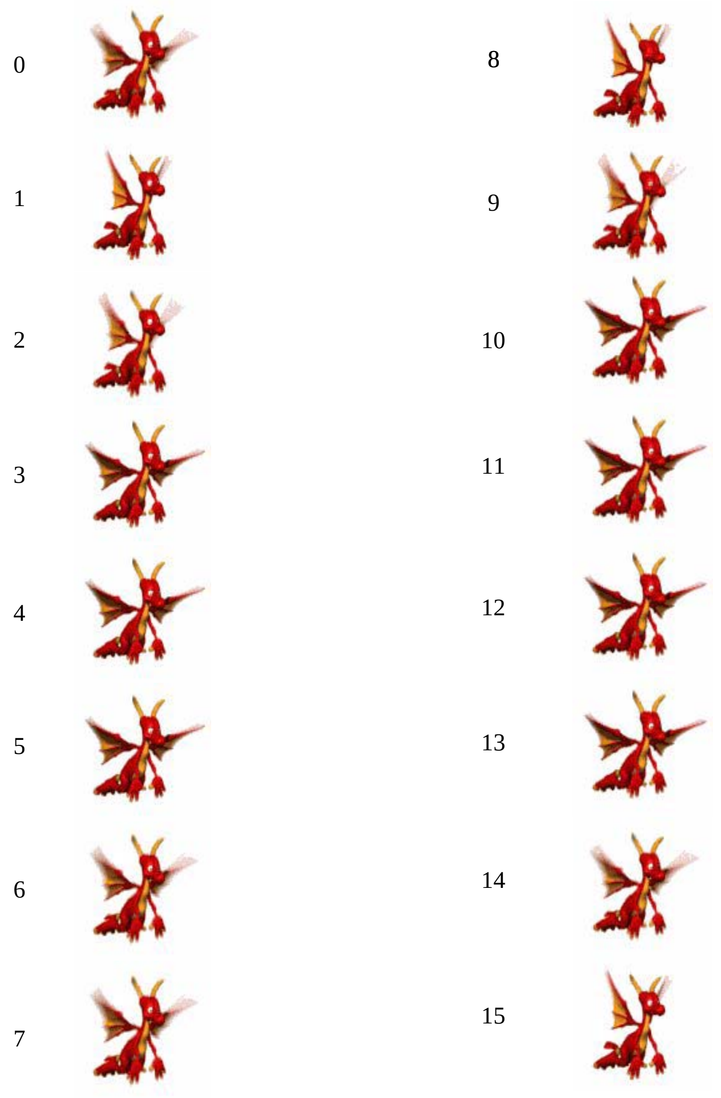


Figure 16-6: Sprite Sequencing Example

16.5 Reference Point Based 90°, 180° and 270° Rotation + Mirror

Each sprite can be independently rotated (90°, 180°, 270°) and/or mirrored. The resulting orientation of the sprite is independent of the main display orientation.

Each sprite has a programmable rotation reference point. Unlike other designs where the rotation is always based on the center of the image, this design allows the user to program any point on the display as the rotation axis. This reference point can even be outside of the sprite area.

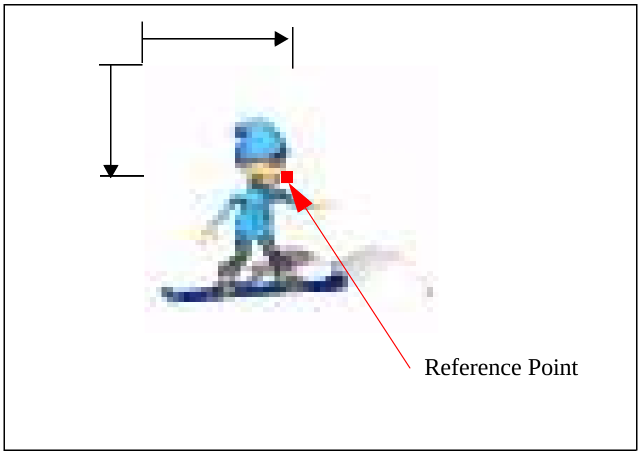


Figure 16-7: Sprite Reference Point

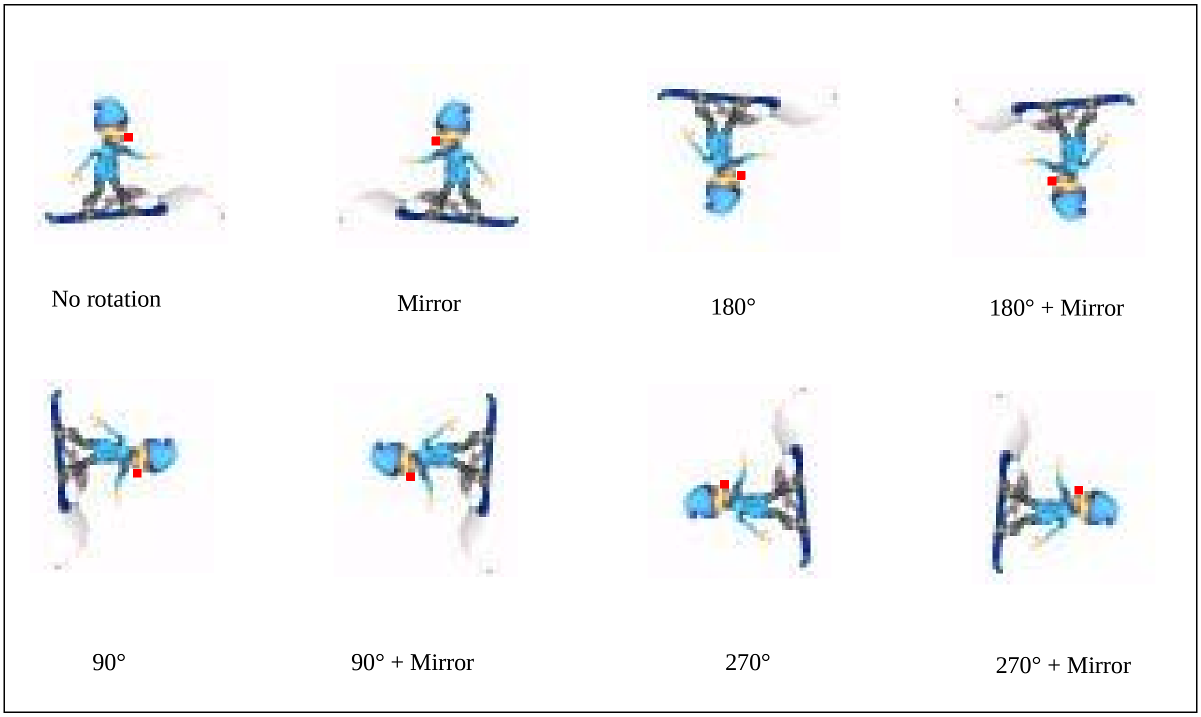


Figure 16-8: Sprite Rotation and Mirror Examples

16.6 Sprite Display Orientation and Positioning

The sprite frame rendered to the SDRAM frame buffer is determined by the dimensions of Sprite #0. Therefore, Sprite #0 defines the resulting SDRAM memory size, and sprite #1 - #15 position is rendered with reference to the rectangle defined by the Sprite #0 frame width and height parameters.

The Main or PIP$^+$ window dimensions and memory start address should match the sprite #0 frame dimensions and the sprite frame buffer start address, in order to display the rendered sprite frame to the Main or PIP$^+$ Window of the display.

The following figures demonstrate how to size and position a sprite for rendering to the frame buffer. Examples are shown for several combinations of rotation and mirroring. In the examples, the large rectangle represents the display area and the small rectangle represents the sprite frame. The figures assume the following values:

A = X offset of the reference point relative to the upper left corner of the sprite
B = Y offset of the reference point relative to the upper left corner of the sprite
C = X offset of the sprite position (reference point) relative to the upper left corner of the display
D = Y offset of the sprite position (reference point) relative to the upper left corner of the display
E = New effective X-Start of the sprite on the display after rotation/mirroring
F = New effective Y-Start of the sprite on the display after rotation/mirroring
G = Frame width of the sprite - A
H = Frame height of the sprite - B
I = New effective X-End of the sprite on the display after rotation/mirroring
J = New effective Y-End of the sprite on the display after rotation/mirroring

Note
If the frame size of the sprite is smaller than the virtual size, the top left portion of the image in memory is displayed. For 90° or 270°, the bottom left portion of the original non-rotated image is displayed since that part is the top left of the rotated image.

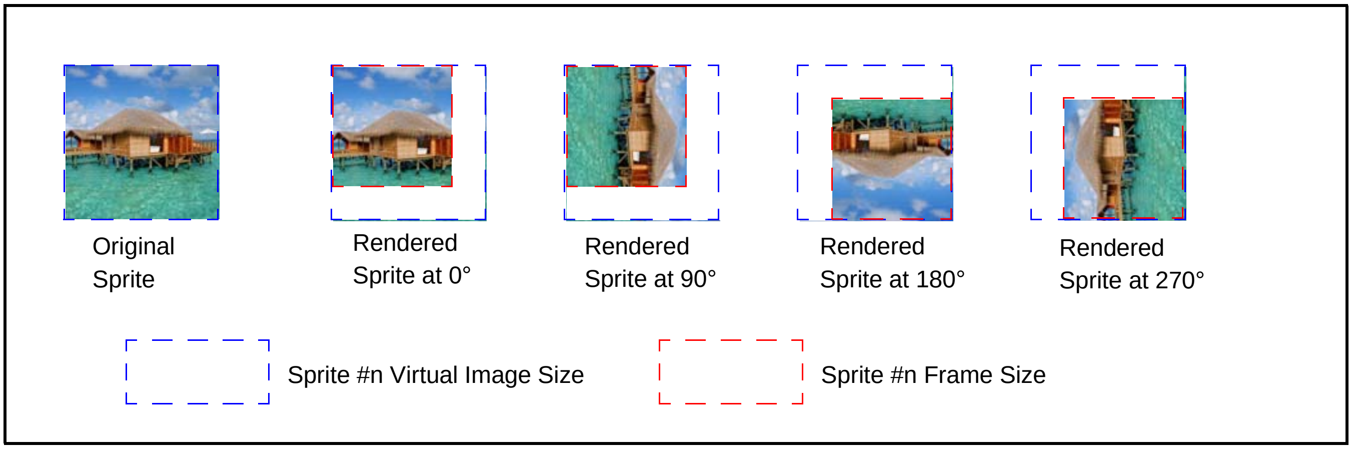


Figure 16-9: Sprite Frame Size Less Than Sprite Virtual Size

0° Rotation with Mirror Disabled

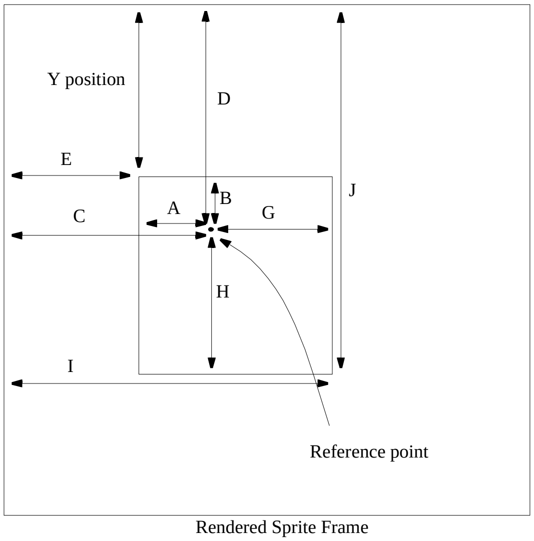


Figure 16-10: Sprite Display for Rotation 0° with Mirror Disabled

$E = C - A$ $\qquad$ $F = D - B$

$I = C + G$ $\qquad$ $J = D + H$

90° Rotation with Mirror Disabled

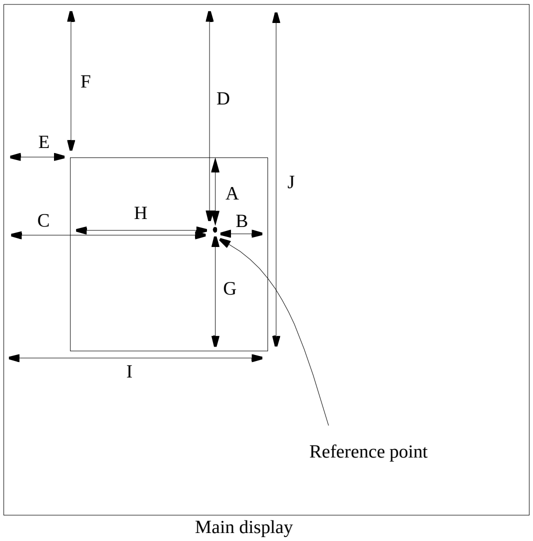


Figure 16-11: Sprite Display for Rotation 90° with Mirror Disabled

$E = C - H$ $\qquad$ $F = D - A$

$I = C + B$ $\qquad$ $J = D + G$

180° Rotation with Mirror Disabled

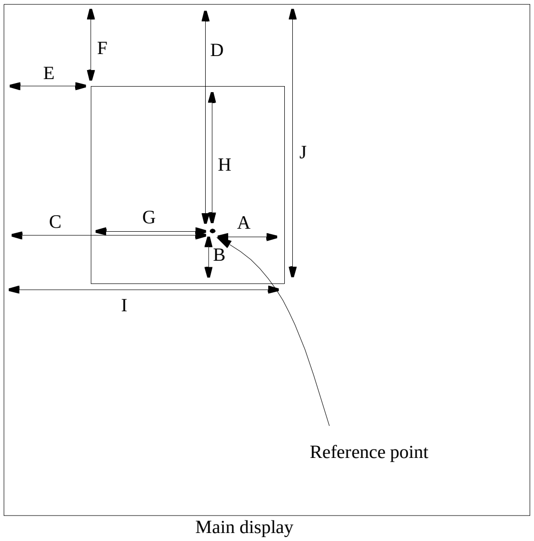


Figure 16-12: Sprite Display for Rotation 180° with Mirror Disabled

$E = C - G$ $F = D - H$

$I = C + A$ $J = D + B$

270° Rotation with Mirror Disabled

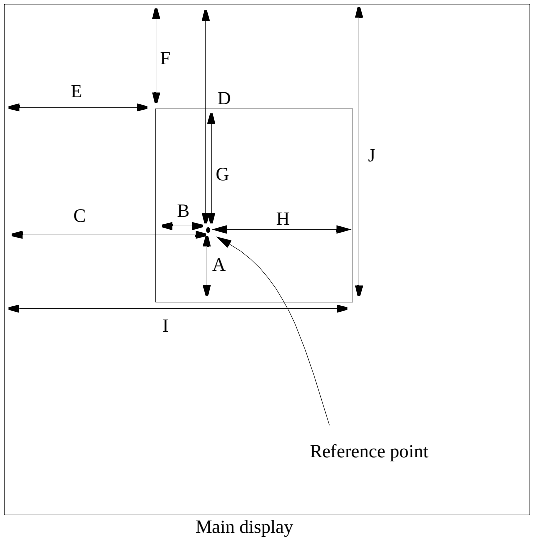


Figure 16-13: Sprite Display for Rotation 270° with Mirror Disabled

$E = C - B$ $F = D - G$

$I = C + H$ $J = D + A$

0° Rotation with Mirror Enabled (Left <-> Right)

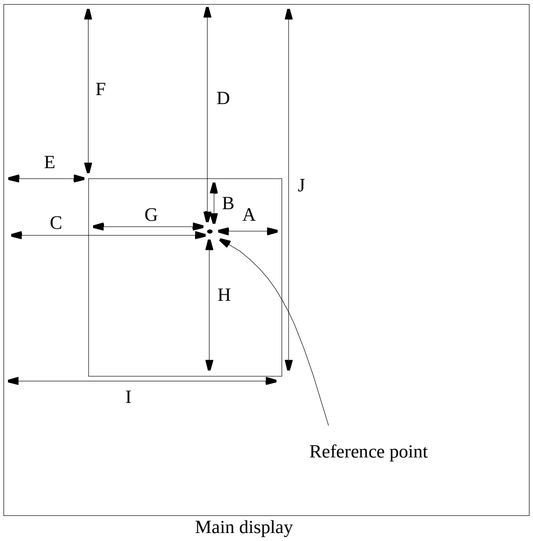


Figure 16-14: Sprite Display for Rotation 0° with Mirror Enabled

E = C - G	F = D - B
I = C + A	J = D + H

90° Rotation with Mirror Enabled

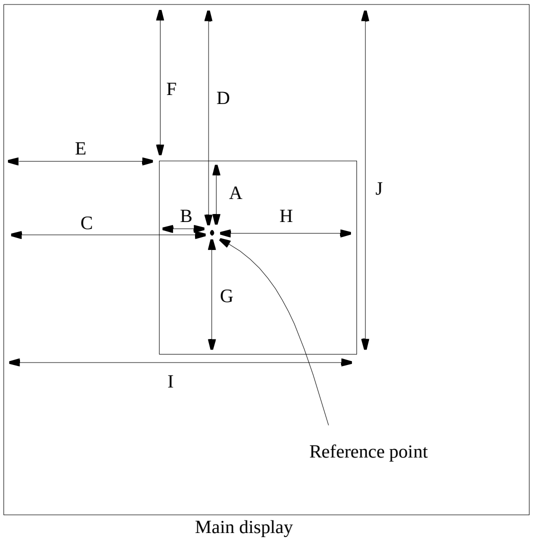


Figure 16-15: Sprite Display for Rotation 90° with Mirror Enabled

E = C - B	F = D - A
I = C + H	J = D + G

180° Rotation with Mirror Enabled

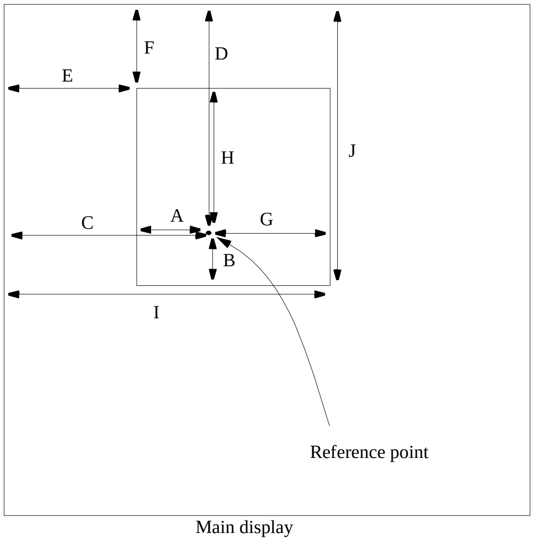


Figure 16-16: Sprite Display for Rotation 180° with Mirror Enabled

E = C - A   F = D - H

I = C + G   J = D + B

270° Rotation with Mirror Enabled

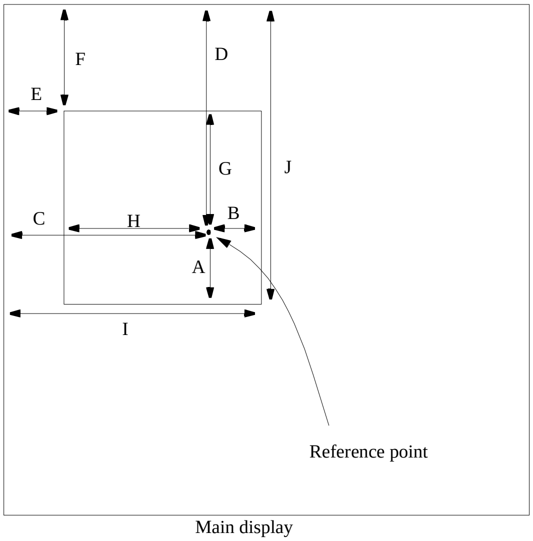


Figure 16-17: Sprite Display for Rotation 270° with Mirror Enabled

E = C - H   F = D - G

I = C + B   J = D + A

16.7 Arbitrary Angle Rotation

16.7.1 Sprite Example with Arbitrary Rotation Disabled

The following example of configures the Sprite Parameters for a 4x4 sprite image with Arbitrary Rotation disabled, REG[1xxxh +00h] bit 7 = 0b.

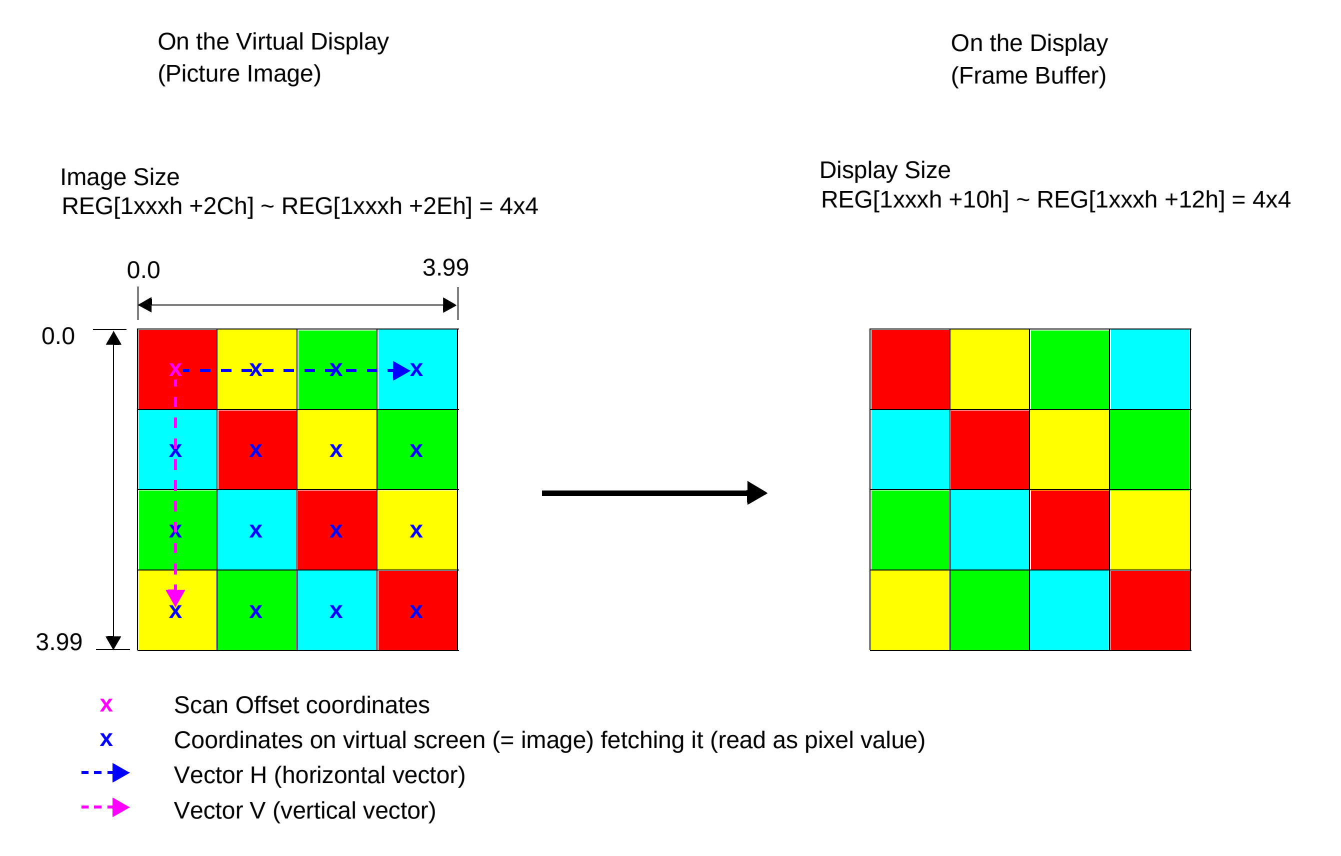

On the Virtual Display
(Picture Image)
On the Display
(Frame Buffer)
Image Size
REG[1xxxh +2Ch] ~ REG[1xxxh +2Eh] = 4x4
Display Size
REG[1xxxh +10h] ~ REG[1xxxh +12h] = 4x4
0.0
3.99
0.0
3.99
x
x
--▶
--▶
Scan Offset coordinates
Coordinates on virtual screen (= image) fetching it (read as pixel value)
Vector H (horizontal vector)
Vector V (vertical vector)

Figure 16-18: Sprite Example with Arbitrary Rotation Disabled

When arbitrary rotation is disabled (REG[1xxxh +00h] bit 7 = 0b), setting the X,Y Scan Vector/Offset registers (REG[1xxxh +30h] ~ REG[1xxxh + 46h]) is not required. These values are generated internally before the sprite operation is done.

To perform the sprite operation, the sprite engine fetches the pixel data from x, and starts drawing to the frame buffer. Then, it fetches the pixel data for the next pixel defined in Vector H (horizontal vector), and draws it to the frame buffer. Once the sprite has drawn 4 pixels as defined by the display X size, the fetch coordinates are moved according to Vector V (vertical vector).

The sprite engine continues fetching the next pixel data and drawing it to the frame buffer until the Y size is reached.

The following table summarizes the actual values that would be programming for the above example.

Table 16-1: Sprite Example with Arbitrary Rotation Disabled

Item	Parameter Value	Register Setting
x	X Scan Offset = 0.5 Y Scan Offset = 0.5	REG[1xxxh +42h], REG[1xxxh +40] = 00000100h (0.5 in 1.22.9 format) REG[1xxxh +46h], REG[1xxxh +44] = 00000100h (0.5 in 1.22.9 format)
- - -▶	X Scan Vector H = 1.0 Y Scan Vector H = 0.0	REG[1xxxh +32h], REG[1xxxh +30] = 00000200h (1.0 in 1.22.9 format) REG[1xxxh +36h], REG[1xxxh +34] = 00000000h (0.0 in 1.22.9 format)
- - -▶	X Scan Vector V = 0.0 Y Scan Vector V = 1.0	REG[1xxxh +3Ah], REG[1xxxh +38] = 00000000h (0.0 in 1.22.9 format) REG[1xxxh +3Eh], REG[1xxxh +3C] = 00000200h (1.0 in 1.22.9 format)
Image Size	Virtual Image Width = 4 Virtual Image Height = 4	REG[1xxxh +2Ch] = 4 REG[1xxxh +2Eh] = 4
Display Size	Frame Width = 4 Frame Height = 4	REG[1xxxh +10h] = 4 REG[1xxxh +12h] = 4

Note

To convert from a fixed zero point value to 1.22.9 format, shift left by 9 bits. For example, to convert 0.5 into 1.22.9 format, do the following.

0.5 x 200h = 256 = 00000100h

16.7.2 Sprite Example with Arbitrary Rotation Enabled

The following example of configures the Sprite Parameters for a 4x4 sprite image with Arbitrary Rotation enabled, REG[1xxxh +00h] bit 7 = 10b.

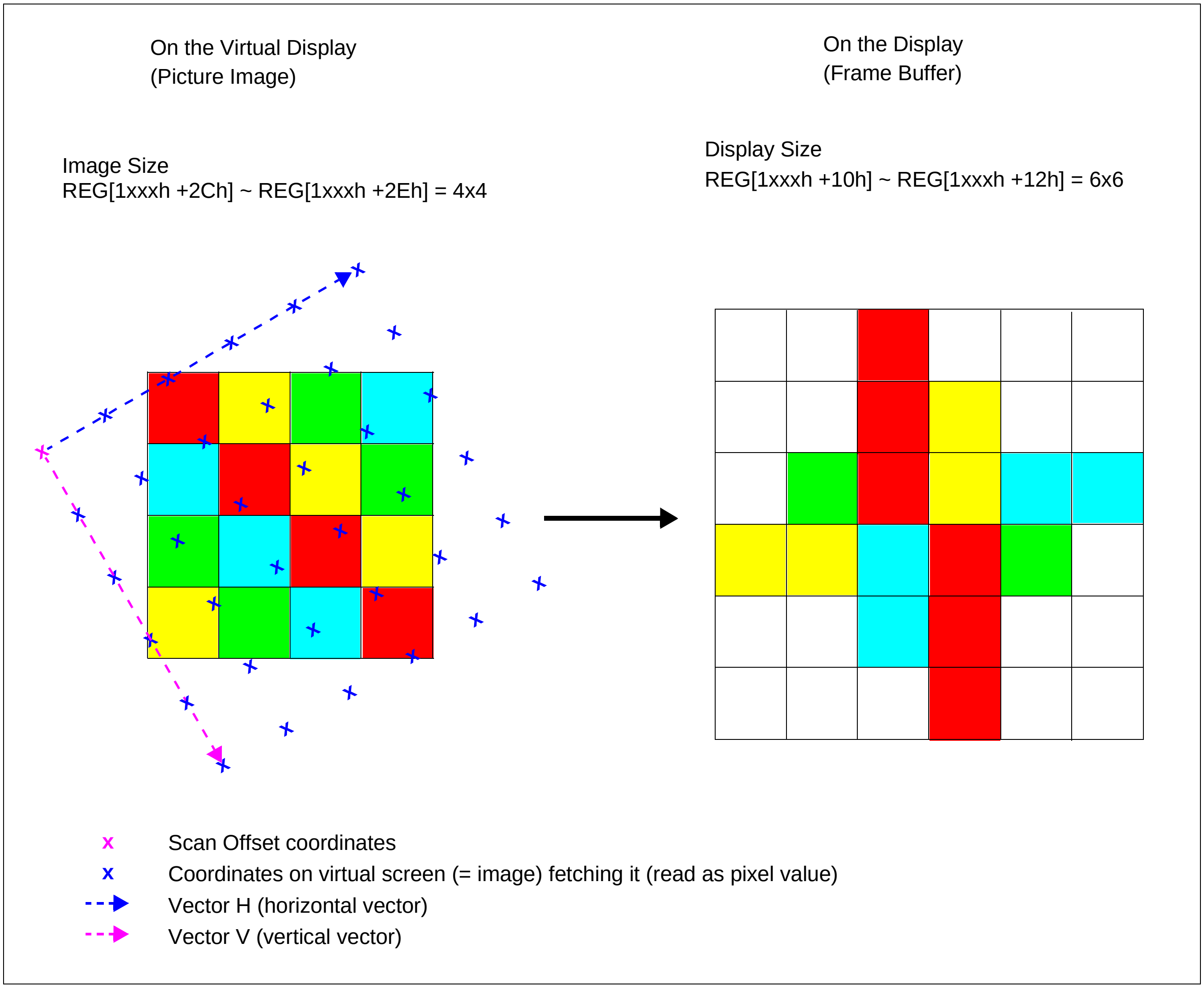

On the Virtual Display
(Picture Image)
On the Display
(Frame Buffer)
Image Size
REG[1xxxh +2Ch] ~ REG[1xxxh +2Eh] = 4x4
Display Size
REG[1xxxh +10h] ~ REG[1xxxh +12h] = 6x6
x Scan Offset coordinates
x Coordinates on virtual screen (= image) fetching it (read as pixel value)
Vector H (horizontal vector)
Vector V (vertical vector)

Figure 16-19: Sprite Example with Arbitrary Rotation Enabled

The above figure shows an example of a 4x4 sprite image with an arbitrary angle right rotation of 30°. Note that rotating such a small image will result in some warping of the image.

When rotating the sprite image, the display size must be expanded to accommodate the entire rotated sprite image. This size is determined based on the required size for the image when rotated by 45°, which results in the largest increase in size of the circumscribed square. The required size can be calculated from the following formulae.

required display width = original sprite width x square root of 2
required display height = original sprite height x square root of 2

For a 4x4 sprite, the following calculation applies to both width and height.

$$4 \times \sqrt{2} \approx 5.66$$

For this example, both the width and height must be rounded up to the next integer value (5.66 -> 6). This means that the display size must be programmed to 6x6. It is important to calculate the minimum required display size rather than over estimate a "safe" size as the length of time taken to draw the rotated sprite image is proportional to the display size.

Table 16-2: Sprite Example with Arbitrary Rotation Enabled

Item	Parameter Value	Register Setting
x	X Scan Offset ≈ 1.415 (see equations below) Y Scan Offset ≈ 1.085 (see equations below)	REG[1xxxh +42h], REG[1xxxh +40] = FFFFFD2Ch REG[1xxxh +46h], REG[1xxxh +44] = 0000022Bh
⇢	X Scan Vector H ≈ 0.866 (see equations below) Y Scan Vector H = -0.5 (see equations below)	REG[1xxxh +32h], REG[1xxxh +30] = 000001BBh REG[1xxxh +36h], REG[1xxxh +34] = FFFFFF01h
⇢	X Scan Vector V = 0.5 (see equations below) Y Scan Vector V ≈ 0.866 (see equations below)	REG[1xxxh +3Ah], REG[1xxxh +38] = 000000FFh REG[1xxxh +3Eh], REG[1xxxh +3C] = 000001BBh
Image Size	Virtual Image Width = 4 Virtual Image Height = 4	REG[1xxxh +2Ch] = 4 REG[1xxxh +2Eh] = 4
Display Size	Frame Width = 6 Frame Height = 6	REG[1xxxh +10h] = 6 REG[1xxxh +12h] = 6

Equations

The following equations are used in calculating the X/Y Scan Vector/Offset values shown in Table 16-2:, "Sprite Example with Arbitrary Rotation Enabled," on page 417.

X Scan Vector H = cos (-θ) ÷ X (Expansion Rate)
Y Scan Vector H = sin (-θ) ÷ Y (Expansion Rate)

X Scan Vector V = -sin (-θ) ÷ X (Expansion Rate)
Y Scan Vector V = cos (-θ) ÷ Y (Expansion Rate)

X Scan Offset = (Image X size ÷ 2) - ((X Scan Vector H x (Display X size - 1)) + (X Scan Vector V x (Display X size - 1))) ÷ 2
Y Scan Offset = (Image Y size ÷ 2) - ((Y Scan Vector H x (Display Y size - 1)) + (Y Scan Vector V x (Display Y size - 1))) ÷ 2

Note

The Expansion Rate is 2.0 for twice the size, and 0.5 for half the size.

The following figure visualizes how the X/Y Scan Offset is calculated.

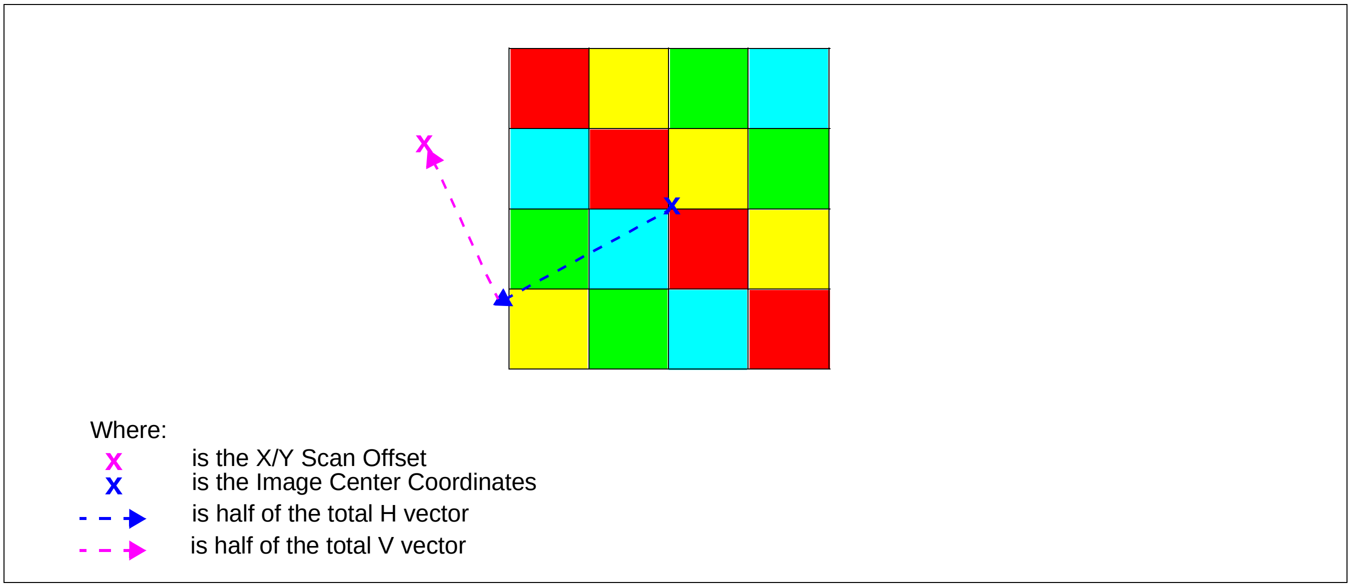

X
X
Where:
X is the X/Y Scan Offset
X is the Image Center Coordinates
- - ➤ is half of the total H vector
- - ➤ is half of the total V vector

Figure 16-20: X/Y Scan Offset Calculation

The above equations allow the rotated image to have an independent expansion ratio for both the H and V directions. When the expansion ratio is different between H and V, the image is transformed into the rectangle. Depending on the equation, it can be transformed into either a diamond or a parallelogram shape. For example, if the following equations will create a diamond shape. The parameters that change from the original equations are highlighted in red.

X Scan Vector H = cos (-θ) ÷ X (Expansion Rate)
Y Scan Vector H = sin (-θ) ÷ X (Expansion Rate)

X Scan Vector V = -sin (-θ) ÷ Y (Expansion Rate)
Y Scan Vector V = cos (-θ) ÷ Y (Expansion Rate)

X Scan Offset = (Image X size ÷ 2) - ((X Scan Vector H x (Display X size - 1)) + (X Scan Vector V x (Display X size - 1))) ÷ 2
Y Scan Offset = (Image Y size ÷ 2) - ((Y Scan Vector H x (Display Y size - 1)) + (Y Scan Vector V x (Display Y size - 1))) ÷ 2

16.8 Programming Flow

The general procedure for programming the sprite engine is as follows:

1. Configure the Sprite Color Format
2. Configure the Sprite Trigger Control
3. Configure the Sprite Interrupt
4. Configure the frame sequence length and trigger count for each sprite
5. Setup the Sprite Frame Buffer Start Address
6. For each enabled sprite, setup:
    a. Z-order, Rotation, Mirror
    b. Transparency and specify the transparent key color (if applicable)
    c. Specify the sprite SDRAM start address (and rotation if enabled)
    d. Specify the sprite frame size, location, and reference point
    e. Load the sprite frame data into SDRAM

Once the sprite module is enabled and the respective sprite data is loaded into SDRAM, the sprite data can be rendered to the frame buffer in SDRAM by initiating a manual sprite paint, or automatically painted based on the TFT vertical sync signal. For manual trigger mode, a sprite paint is triggered using REG[1704h] bit 0. For non-manual trigger mode, the sprites will update based on trigger count and sprite paint trigger settings.

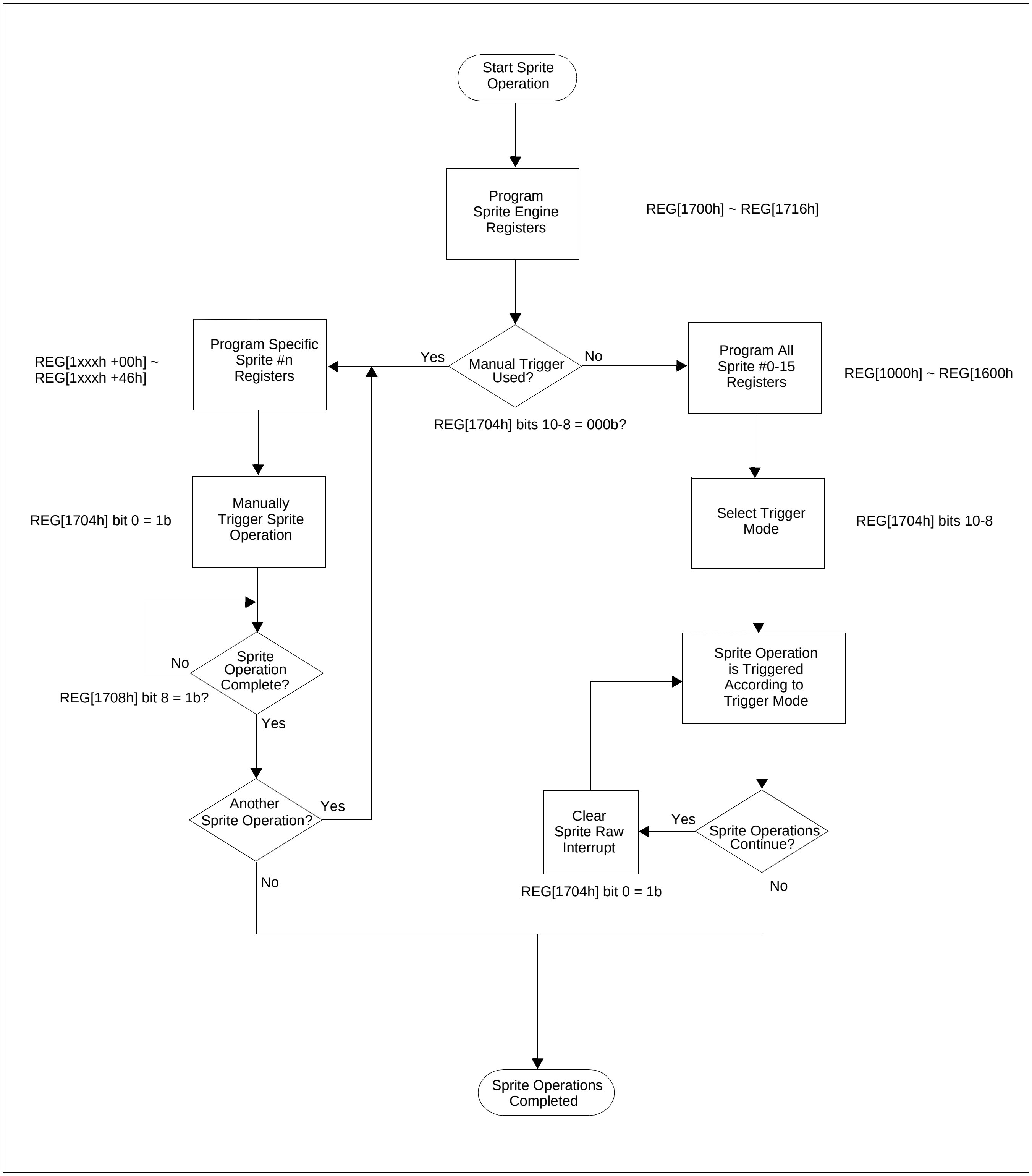

Start Sprite Operation
Program Sprite Engine Registers
REG[1700h] ~ REG[1716h]
Manual Trigger Used?
REG[1704h] bits 10-8 = 000b?
Yes
No
Program Specific Sprite #n Registers
REG[1xxxh +00h] ~ REG[1xxxh +46h]
Program All Sprite #0-15 Registers
REG[1000h] ~ REG[1600h
Manually Trigger Sprite Operation
REG[1704h] bit 0 = 1b
Select Trigger Mode
REG[1704h] bits 10-8
Sprite Operation Complete?
REG[1708h] bit 8 = 1b?
No
Yes
Sprite Operation is Triggered According to Trigger Mode
Another Sprite Operation?
Yes
No
Clear Sprite Raw Interrupt
REG[1704h] bit 0 = 1b
Sprite Operations Continue?
Yes
No
Sprite Operations Completed

Figure 16-21: Sprite Operation Flow

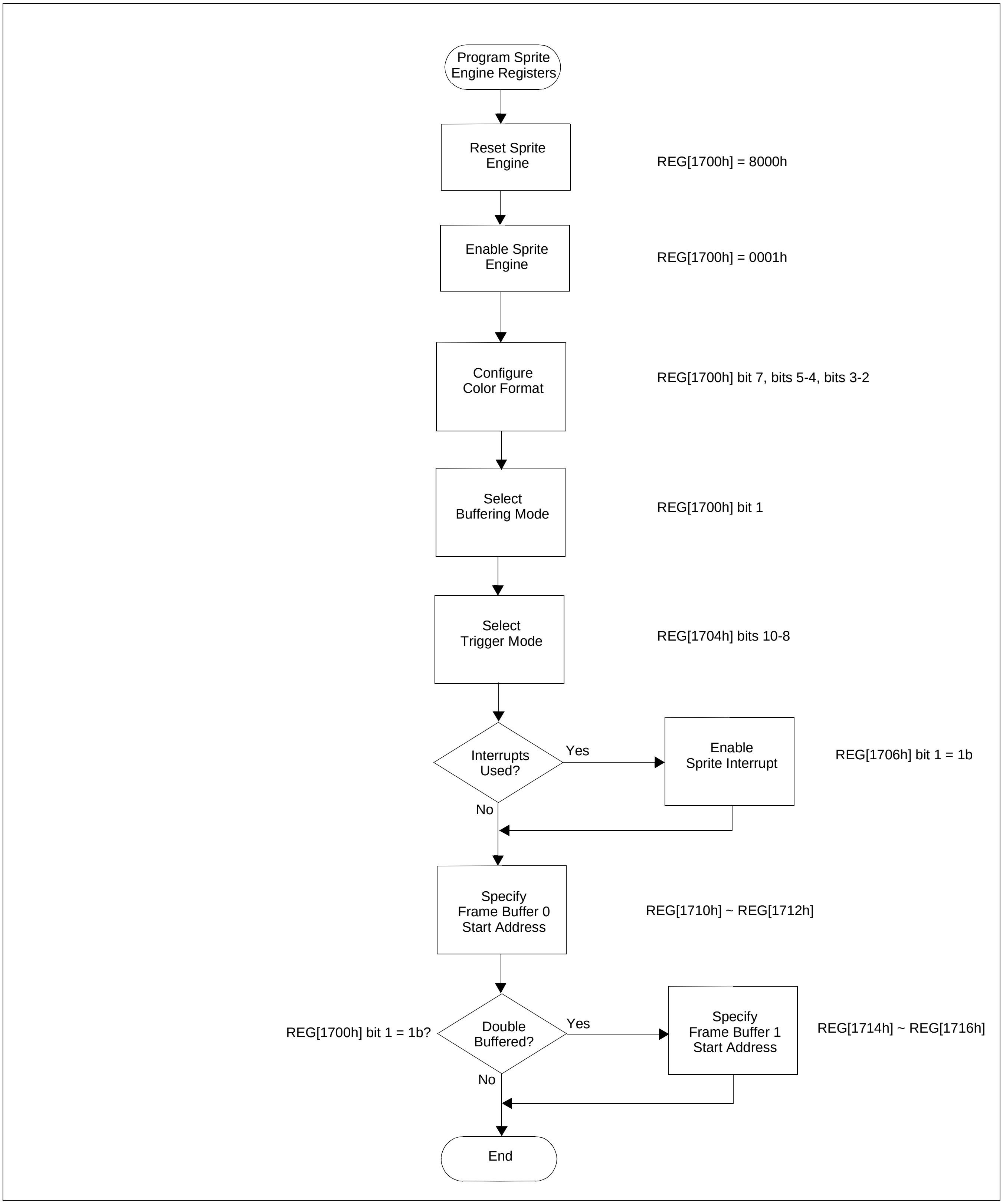


Figure 16-22: Programming Sprite Engine Registers Flow

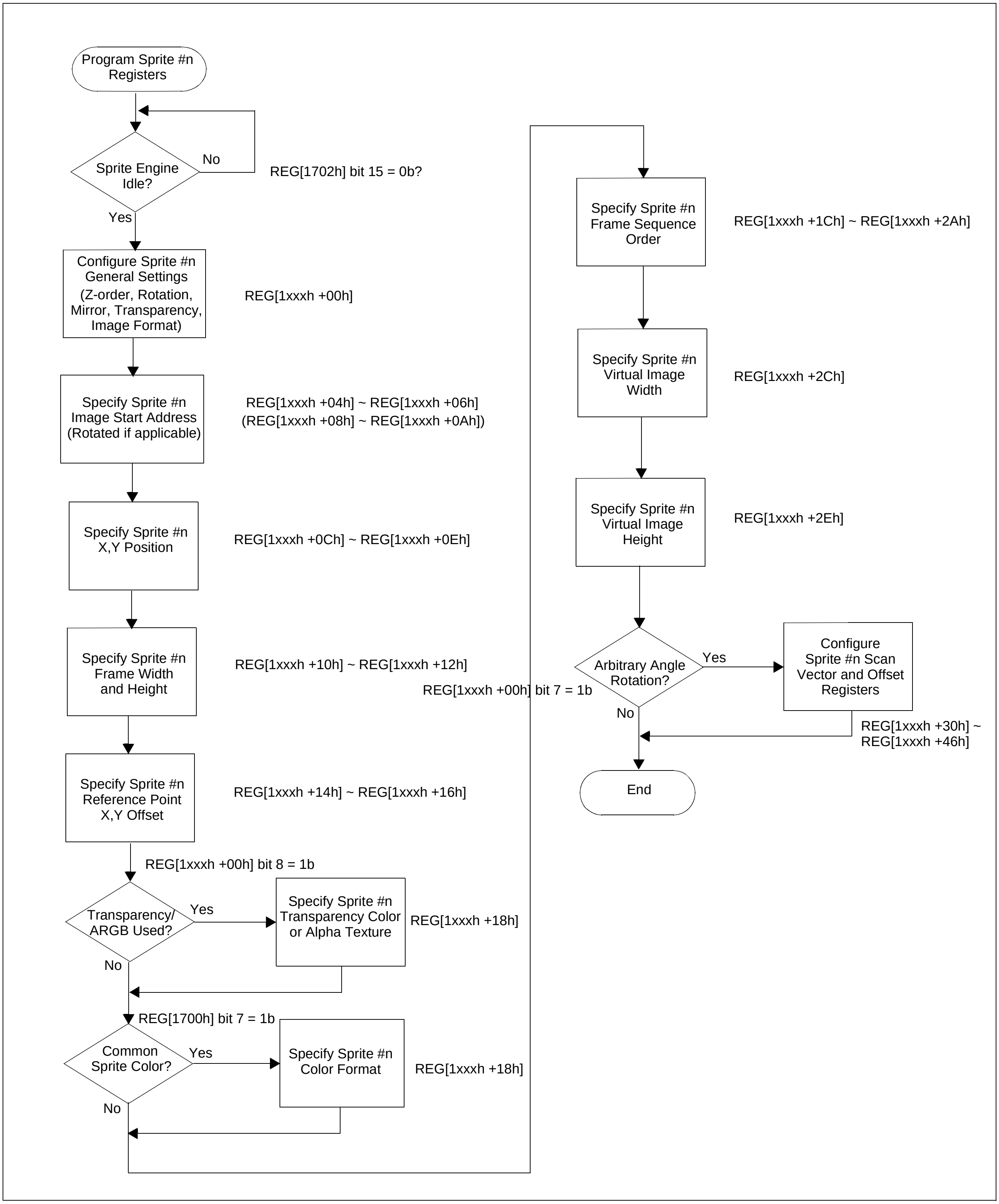

Program Sprite #n Registers
Sprite Engine Idle?
No
Yes
REG[1702h] bit 15 = 0b?
Configure Sprite #n General Settings (Z-order, Rotation, Mirror, Transparency, Image Format)
REG[1xxxh +00h]
Specify Sprite #n Image Start Address (Rotated if applicable)
REG[1xxxh +04h] ~ REG[1xxxh +06h]
(REG[1xxxh +08h] ~ REG[1xxxh +0Ah])
Specify Sprite #n X,Y Position
REG[1xxxh +0Ch] ~ REG[1xxxh +0Eh]
Specify Sprite #n Frame Width and Height
REG[1xxxh +10h] ~ REG[1xxxh +12h]
Specify Sprite #n Reference Point X,Y Offset
REG[1xxxh +14h] ~ REG[1xxxh +16h]
REG[1xxxh +00h] bit 8 = 1b
Transparency/ ARGB Used?
Yes
No
Specify Sprite #n Transparency Color or Alpha Texture
REG[1xxxh +18h]
REG[1700h] bit 7 = 1b
Common Sprite Color?
Yes
No
Specify Sprite #n Color Format
REG[1xxxh +18h]
Specify Sprite #n Frame Sequence Order
REG[1xxxh +1Ch] ~ REG[1xxxh +2Ah]
Specify Sprite #n Virtual Image Width
REG[1xxxh +2Ch]
Specify Sprite #n Virtual Image Height
REG[1xxxh +2Eh]
Arbitrary Angle Rotation?
REG[1xxxh +00h] bit 7 = 1b
Yes
No
Configure Sprite #n Scan Vector and Offset Registers
REG[1xxxh +30h] ~
REG[1xxxh +46h]
End

Figure 16-23: Programming the Sprite #n Registers Flow

16.9 Image Format Conversion

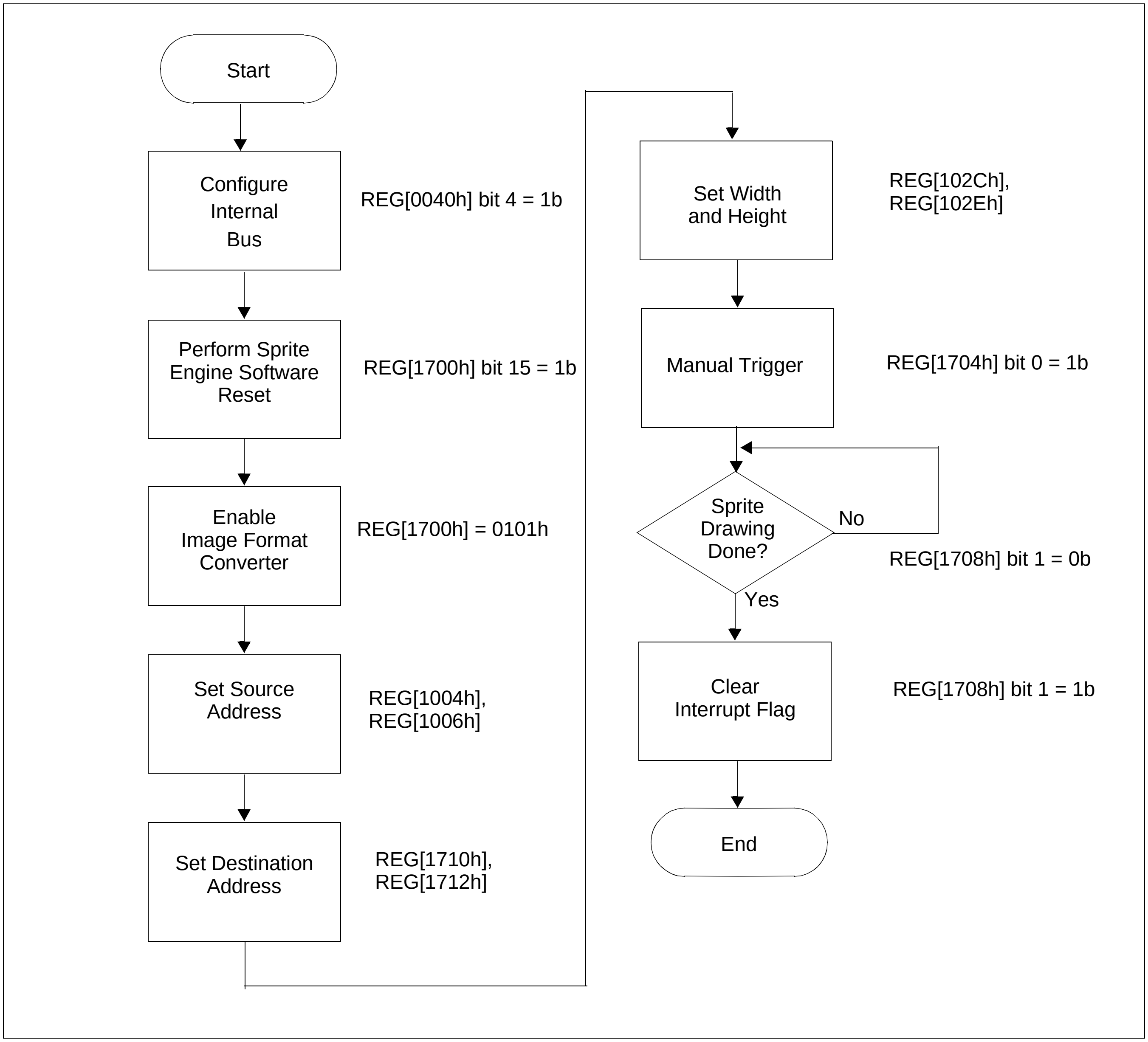


Figure 16-24: Image Format Conversion Example Sequence

Note

When an image format conversion is performed, confirm that the conversion is complete (see REG[1704h] bit 1) before starting a new image format conversion.

17 Command FIFO

The S1D13513 includes a Command FIFO that is designed to reduce CPU load when issuing multiple Sprite or BitBLT commands. By using the Command FIFO, software allows the Command FIFO logic to internally handle the delays inherent with some commands (i.e. start BitBLT operation). This frees the software, and thus the Host, from the extra load required by some commands.

Note

When using the Sprite or BitBLT engine for the first time, Clock Enable for Command FIFO (REG[0462h] bit 5) has to be enabled before any command write or read access to this Command FIFO, then program the registers directly. For subsequent commands/parameters, place them in the Command FIFO to allow the Command FIFO logic to automatically handle the "busy" situations inherent with Sprite and BitBLT operations.

The Command FIFO can contain up to a maximum of 64 command entries which can be mixed between commands to the Sprite Engine and commands to the BitBLT engine. However, the Sprite Engine and BitBLT engine must not be operating simultaneously. For example, the Sprite Engine can be performing sprite operations while the BitBLT Engine is being programmed, but it cannot be drawing sprites at the same time as the BitBLT Engine is performing BitBLT operations.

The Command FIFO is accessed by writing to the register space from REG[4000h] to REG[4FFFh]. This space is mapped using the same offset addresses as the Sprite registers from REG[1000h] through REG[17FFh] and the BitBLT registers from REG[1800h] through REG[1FFFh].

Reading any register within the Command FIFO register space (REG[4000h] ~ REG[4FFFh]) will return the number of entries currently available in the Command FIFO. When the Command FIFO is empty, a read will return 40h (64). When the Command FIFO is being used, it will return (64 - the number of used entries).

When a register within the range of REG[4000h] through REG[4FFFh] is written, it is translated into a "command" and placed into the Command FIFO. Each command consists of three parts as follows.

Table 17-1 : Command Structure

1-bit	10-bit	16-bit
Sprite/BitBLT	Address	Data

The first bit defines whether the command is for the Sprite engine (bit = 1) or the BitBLT engine (bit = 0). The 10-bit address specifies which register will be programmed when the appropriate engine becomes available. Lastly, the data portion contains the value that will be written to the specified register.

When either the Sprite engine or the BitBLT engine is idle, the next command destined for that engine is performed. This writes the selected data value into the specified register index as stored in the Command FIFO. Some commands may take a short while to complete (i.e. starting a BitBLT operation). During this time, the Command FIFO automatically waits for the issued command to finish before it performs the next command for that engine.

The following diagram shows the paths that can be used to program the Sprite and BitBLT engine registers. The registers can be programmed directly which requires the software to manage the delays for some "commands" using polling or interrupts. Or the registers can be programmed using the Command FIFO which allows "commands" to be queued internally and relieves software of the requirement to manage delays.

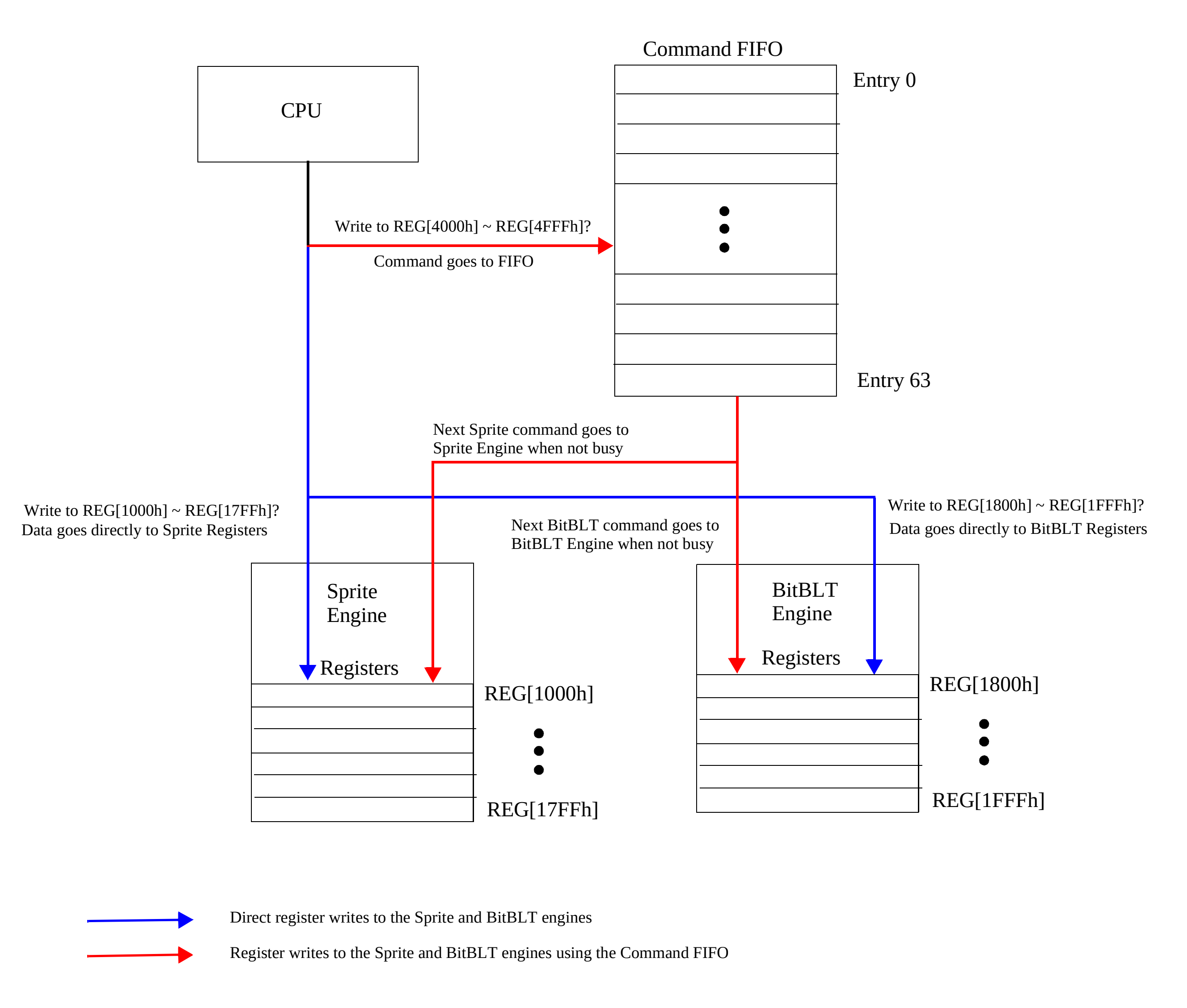

Command FIFO
CPU
Entry 0
Write to REG[4000h] ~ REG[4FFFh]?
Command goes to FIFO
Entry 63
Next Sprite command goes to
Sprite Engine when not busy
Write to REG[1000h] ~ REG[17FFh]?
Data goes directly to Sprite Registers
Write to REG[1800h] ~ REG[1FFFh]?
Data goes directly to BitBLT Registers
Next BitBLT command goes to
BitBLT Engine when not busy
Sprite
Engine
Registers
REG[1000h]
REG[17FFh]
BitBLT
Engine
Registers
REG[1800h]
REG[1FFFh]
Direct register writes to the Sprite and BitBLT engines
Register writes to the Sprite and BitBLT engines using the Command FIFO

Figure 17-1: Command FIFO Example

18 SDRAM Interface

The S1D13513 SDRAM is designed to use external SDRAM or mobile SDRAM. It supports x16 and x32 SDRAM interfaces which allows for SDRAM sizes of 8/16/32/64M bytes (64/128/256/512Mbit). The S1D13513 includes no embedded memory.

Note

The x32 SDRAM interface and 32/64M byte sizes of SDRAM are only available for the PBGA package.

The S1D13513 uses the external SDRAM for the display buffer and some 2D BitBLT/Sprite functions. It is addressable through direct or indirect access modes. For details on addressing the memory, refer to Section 8, “Memory Map” on page 121.

18.1 SDRAM Initialization

The SDRAM must be initialized using the SDRAM Configuration registers (REG[1C00h] ~ REG[1C14h]) before using the SDRAM. The initialization sequence is differs depending on whether normal SDRAM or Mobile SDRAM is used. The recommended initialization sequence for each type of SDRAM is included in the next sections.

18.1.1 Initializing Normal SDRAM

The following sequence should be used to initialize normal SDRAM. For details on specific SDRAM requirements, refer to the SDRAM specification.

1. Wait at least 100µs after SDRAM Power Up. (For details, refer to the SDRAM specification.)

2. Configure Memory Configuration Register 1 (REG[1C04h]) according to the requirements of the SDRAM used.

3. Configure Memory Configuration Register 2 (REG[1C06h]) according to the SDRAM memory size and clock configuration.

4. Configure Memory Configuration Register 0 (REG[1C02h]) according to the SDRAM bus width. The SDRAM can be enabled (REG[1C02h] bit 0) during the same write. Enabling the SDRAM starts the SDRAM initialization process.

5. Wait until the Memory Initialized bit returns a 1b (REG[1C02h] bit 7 = 1b).

6. The SDRAM is now ready for use.

18.1.2 Initializing Mobile SDRAM

The following sequence should be used to initialize mobile SDRAM. For details on specific mobile SDRAM requirements, refer to the Mobile SDRAM specification.

1. Wait at least 100µs after Mobile SDRAM Power Up. (For details, refer to the Mobile SDRAM specification.)
2. Configure Memory Configuration Register 1 (REG[1C04h]) according to the requirements of the Mobile SDRAM used.
3. Configure Memory Configuration Register 2 (REG[1C06h]) according to the Mobile SDRAM memory size and clock configuration.
4. Configure Mobile SDRAM Configuration Register (REG[1C12h]) depending on the Mobile SDRAM extended mode usage. Bit 7 of REG[1C12h] must be set to 1b when Mobile SDRAM is used.
5. Configure Memory Configuration Register 0 (REG[1C02h]) according to the Mobile SDRAM bus width. The Mobile SDRAM can be enabled (REG[1C02h] bit 0) during the same write. Enabling the Mobile SDRAM starts the initialization process.
6. Wait until the Memory Initialized bit returns a 1b (REG[1C02h] bit 7 = 1b).
7. The Mobile SDRAM is now ready for use.

18.2 Memory Bandwidth

There are many S1D13513 modules that require access to the SDRAM. If too many S1D13513 modules are required for a given implementation, there may not be enough total bandwidth for all modules to have adequate access to the SDRAM. This situation may result in corruption of the display.

The amount of available bandwidth differs between 16-bit and 32-bit memory interface configurations. The following guidelines list the maximum color depth (bpp) supported for each listed resolution when minimal modules are required.

Table 18-1: Resolution vs. Color Depth Guidelines

Resolution	**Color Depth**	
	16-Bit Memory Interface	**32-bit Memory Interface**
1024x768	16 bpp (see Note)	16 bpp
800x600	16 bpp (see Note)	16 bpp
640x480	16 bpp	32 bpp

Note

For 1024x768 and 800x600 resolution displays, only the Main window can be used. If PIP window support is required, the 32-bit memory interface should be used.

Note that for implementations where multiple modules are required (i.e. Camera interface, 2D BitBLT, and Sprite), the resolutions listed above may not be attainable. If a specified resolution and color depth is required and display corruption is visible, increase the Horizontal Total (HT) parameter to the maximum allowed by the panel. If this step does not eliminate the display corruption, it may be necessary to stop using a module with a higher priority than the LCD Controller. The S1D13513 gives access priority to each module as follows:

1. DMA (highest priority)
2. Camera Interface
3. Host Interface
4. LCD Controller
5. 2D BitBLT Operations
6. Sprite Operations (lowest priority)

19 Pulse Width Modulation (PWM)

19.1 PWM Circuit Overview

The PWM circuit operates from an approximate 16KHz PWMSRCCLK clock. The clock divider in REG[3402h] allows this PWM clock to be divided by up to 16 to produce the PWM clock. This PWM clock is 16 times the frequency of a PWM cycle, and that PWM cycle has a variable duty cycle as programmed in the PWM Duty Cycle register, REG[340Ch].

The circuit can pulse the LEDs from completely off to the duty cycle programmed in the PWM Duty Cycle register, REG[340Ch]. An LED pulse is created every 128 clocks of the PULSE_clk which is 1/16th the frequency of the PWM clock.

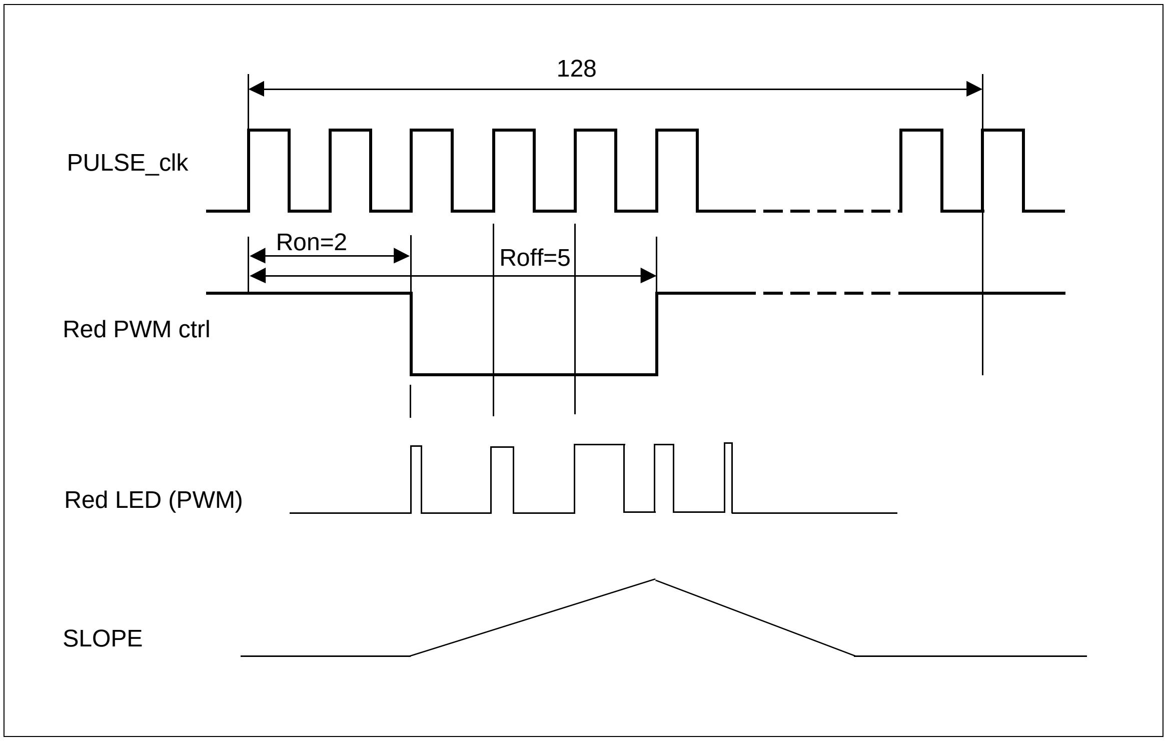


Figure 19-1: PWM Timing Example

In the figure above, the Red on register is programmed to 2 and the Red off register is programmed to 5. If the slope register has a non-zero value, then the PWM duty cycle will ramp up from zero up to the maximum duty cycle programmed in the PWM Duty Cycle register, REG[340Ch].

Notice in the example above, that the PWM circuit may-not have reached its maximum duty cycle (it was still sloping); if the Red off register is programmed with time value that doesn’t allow the slope to fully reach the maximum PWM duty cycle, then it is simply truncated and begins sloping down as shown above. It should also be noted that if the PWM

Slope register (REG[340Ah]) had been programmed with the value “0”, then the PWM output would have gone immediately from completely off, to the full PWM value programmed in the PWM Duty Cycle register (REG[340Ch]) and then to the completely off state (without sloping through the intermediate PWM duty cycle values).

Below is another example where this time, the slope reaches the maximum duty cycle and stays there until the Red off time arrives. It should be noted that all three PWM circuits are synchronized to the 128 clock period and their cycles are in sync with each other. The Red on and Red off settings are used to adjust the three color LED on/off periods relative to each other.

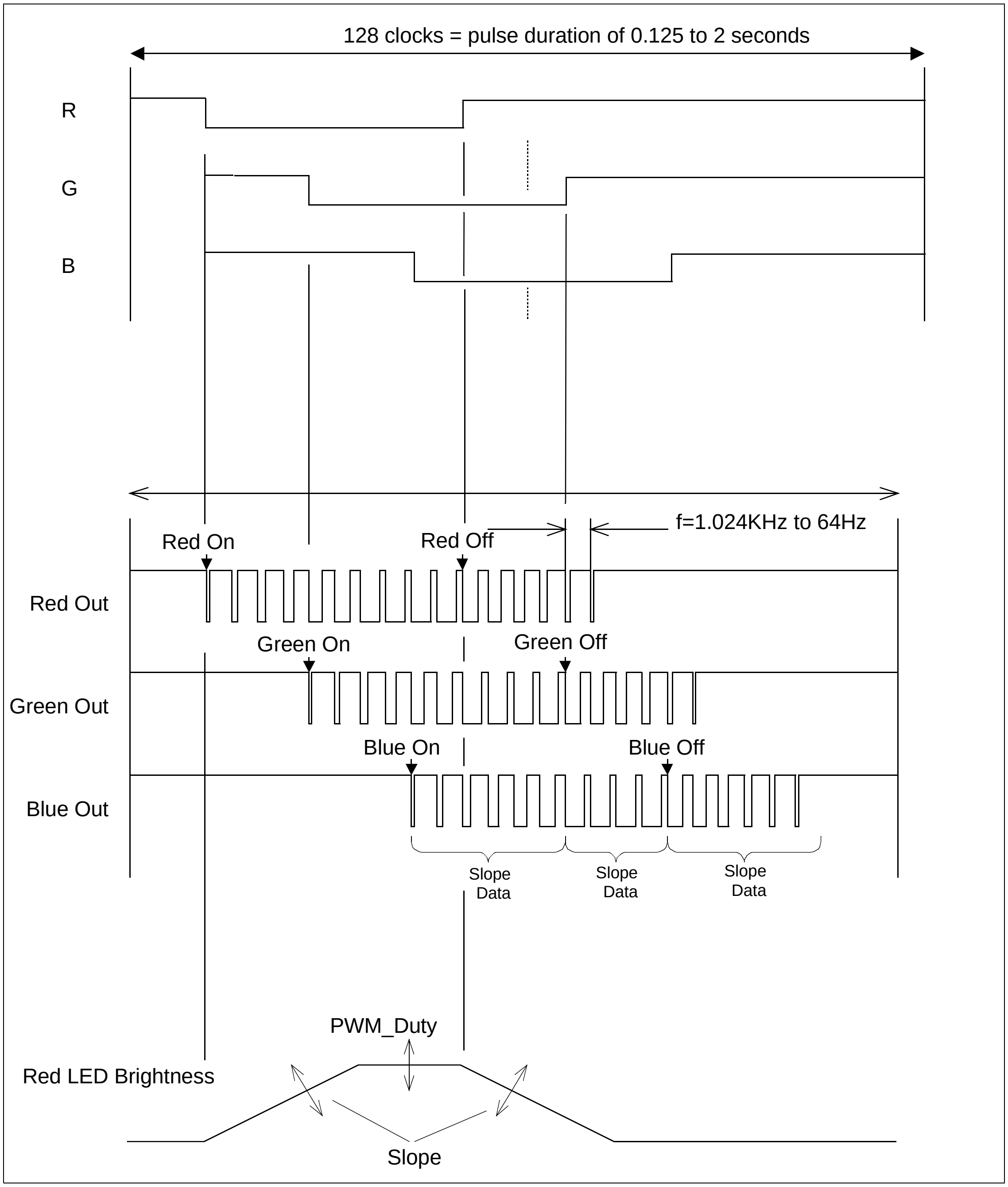


Figure 19-2 PWM Timing Diagram

The White PWM circuit is somewhat different than that of the three color circuits in that it does not have a slope and pulse generation circuit. It is simply turned on to a specific PWM duty cycle or completely turned off.

The clock source for the White PWM circuit is PWMSRCCLK. The period and the duty cycle for the White output are controlled by the White LED Control register, REG[340Eh]. This gives the white PWM output a frequency range of 64Hz to 1Hz, with 64 possible duty cycles.

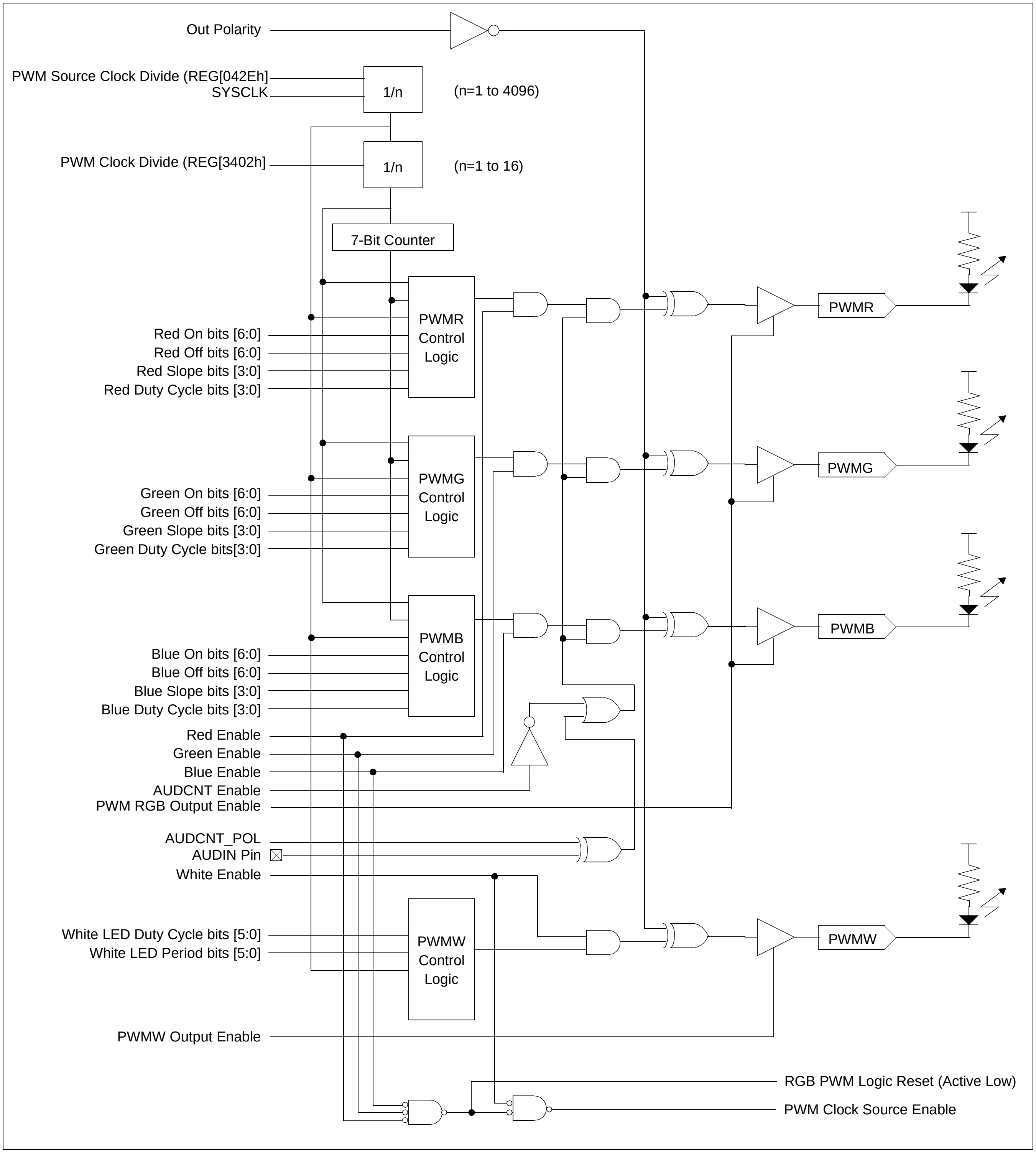


Figure 19-3 PWM Block Diagram

19.2 Other Notes

- The PWM circuit completely shuts down when not in use so users not wanting to make use of it will not suffer any power efficiency degradation when the PWM circuit is idle. For example if REG[3400h] bits 11-8 = 0000b, then the clock to the PWM circuit is completely shut down to prevent any current drain from this circuit.
- HVDD3 is used to supply the output voltage to the output drivers (R,G,B,W)
- All PWM counters and state machines used in the RGB circuit are reset to their initial state when the Red, Green, and Blue PWM enable bits (REG[3400h bits 9, 10, and 11) are ALL turned off (i.e. all three must be zero to reset the PWM reference counters). Since the hardware has no synchronization circuit inside to re-sample the register values that might be being changed by software, it is recommended that the R,G,B PWM circuit be turned off while making changes to the PWM registers. This is not a strict requirement and the circuit will eventually reset to the correct state on the next 128 count cycle, however, strange visual patterns may be seen if the PWM registers are modified mid-cycle while the PWM circuit is operating.
- The AUDIN pin offers further control over the PWM outputs by providing an external input pin. This input pin combined with it’s associated enable bit directly controls the PWM output stage after the PWM circuit.

20 Host Interface

20.1 Hardware Configuration

The S1D13513 is configured using the CNF[8:0] pins which must be connected High through a pull-up resistor or directly to VSS. These configuration pins are used to select the host bus interface type, chip select mode, endian mode, and Clock mode. The state of CNF[8:0] at the rising edge of RESET# determines the configuration of the S1D13513. Changing the state of CNF[8:0] at any other time has no effect.

For a summary of configuration options, see Section 5.3, “Summary of Configuration Options” on page 41.

20.1.1 Bus Type (CNF6)

The S1D13513 supports 2 types of Host bus interface: Host interface bus with bus clock and Host interface bus without bus clock. When CNF6 = 0, Host interfaces with bus clock are selected. When CNF6 = 1, Host interfaces without bus clock are selected. For a complete list of CPU bus types that are supported, refer to Section 5.3, “Summary of Configuration Options” on page 41.

20.1.2 Chip Select (1 CS# vs. 2 CS#)

The S1D13513 supports two types of Chip select mode. Refer to the CNF[5:0] settings for availability of each mode (see Section 5.3, “Summary of Configuration Options” on page 41).

For 1CS# mode, the CS# pin is used for chip select of the S1D13513 and the M/R# pin is used for space selection between memory and register space.

For 2CS# mode, the CS# pin is used for S1D13513 register space selection and the M/R# pin is used for S1D13513 memory space selection.

Note

Not all of CPU bus types support 2 CS# mode.

20.1.3 Endian Mode

The S1D13513 supports both big and little endian modes. The endian mode affects the byte lane bus steering of the host data bus.

When CNF6 = 0, each CNF[4:0] combination specifies an Endian mode for the selected Host bus type.

When CNF6 = 1, CNF5 is used to specify the Endian mode as follows.
CNF5 = 0: Little Endian
CNF5 = 1: Big Endian

For a summary, see Section 5.3, “Summary of Configuration Options” on page 41.

Note

When a Big Endian host interface is selected, the memory accesses are byte swapped. Register accesses are not swapped. Therefore, the registers must be accessed using a method which “byte-swaps” the upper and lower data byte in each register. For details on this requirement, see Section 20.6, “Register Accesses for Big Endian Host Interfaces” on page 452.

20.1.4 CNF[4:0]- Host Bus Interface Type

The S1D13513 supports a variety of Host CPU bus types, including Generic bus, ISA bus, MPC555, SH3, SH4, and traditional Mode 80, Mode 68 interfaces.

Mode 80 has two variations that use different combinations of read/write signals (Type 1 and Type 2). Type 1 and Type 2 parallel host interfaces can use either direct or indirect addressing.

When direct addressing is selected, the address is specified with pins AB[20:1]. Indirect addressing specifies the address using an index register.

For a summary of available Host bus interfaces, see Section 5.3, “Summary of Configuration Options” on page 41.

20.1.5 Serial Host Interface Clock Polarity

The serial Host interface can be configured for HVDD1 or HVDD2 with data valid on either the falling or rising edge.

Table 20-1 : Serial Host interface Configuration

CNF[4:0]	HOST Interface type	Valid Edge
10000b	Serial interface at Host VDD (HVDD1)	Data valid on falling edge
10001b	Serial interface at Panel VDD (HVDD2)	Data valid on falling edge
11000b	Serial interface at Host VDD (HVDD1)	Data valid on rising edge
11001b	Serial interface at Panel VDD (HVDD2)	Data valid on rising edge

20.2 Host Bus Time-out Function

The S1D13513 can detect two types of Host Bus Time-out. If enabled, the Host Interrupt Flag (REG[0020h] bit 0) is set when a time-out condition occurs.

20.2.1 Host Read/Write Cycle Time-out

For this type of Host Bus Time-out, a time-out occurs when a Host read or write access to SDRAM exceeds a specified time. This time-out function is supported for both Direct and Indirect host bus interface modes. Both the PCLK Enable (REG[0462h] bit 3) and the HCLK1 Enable (REG[0462h] bit 1) must be enabled for this time-out function to happen.

This Host Bus Time-out is enabled when REG[0024h] bit 7 = 1b. The Host Time-out Value is specified in system clocks and is set using the Host Time-out Value bits, REG[0024h] bits 6-0. If a Host access to SDRAM exceeds this time, the Host Interrupt Flag in REG[0020h] bit 0 is set.

Interrupts can be used to further determine the type of error. Enabling the Memory Read Error and Memory Write Error Interrupts (see REG[0028h] bits 1-0) will set the appropriate flags in REG[0026h] when a Host Time-out occurs. Once the interrupt has been handled, the status flags in REG[0026h] can be cleared by writing a 1b to the corresponding bit.

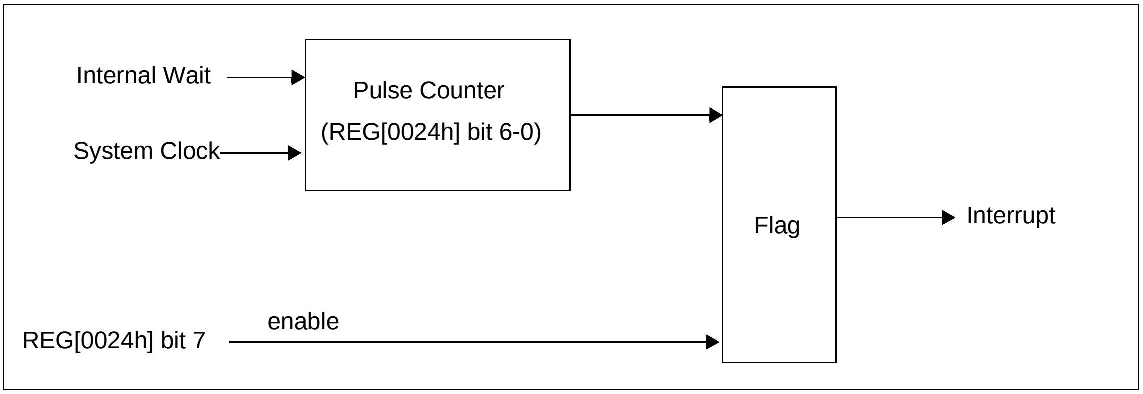


Figure 20-1: Host Read/Write Cycle Time-out Function

20.2.2 Host WAIT# Length Time-out

For this type of Host Bus Time-out, a time-out occurs if the S1D13513 holds WAIT# for more than 2000 Source Clocks. This time-out function is supported for both Direct and Indirect host bus interface modes. The Source Clock is selected using the CNF[8:7] pins.

This Host Bus Time-out is disabled by default. However, the S1D13513 can be configured to use this function to automatically reset when this time-out occurs by setting the Bus Time-out Reset Disable bit to 0b (REG[0472h] bit 0 = 0b). When enabled, the Bus Time-out Reset Interrupt Flag is set in REG[0472h] bit 2 if the time-out condition occurs and the interrupt is enabled (see REG[0472h] bit 1). The flag can be cleared by disabling the Bus Time-out Reset Interrupt (REG[0472h] bit 1 = 1b).

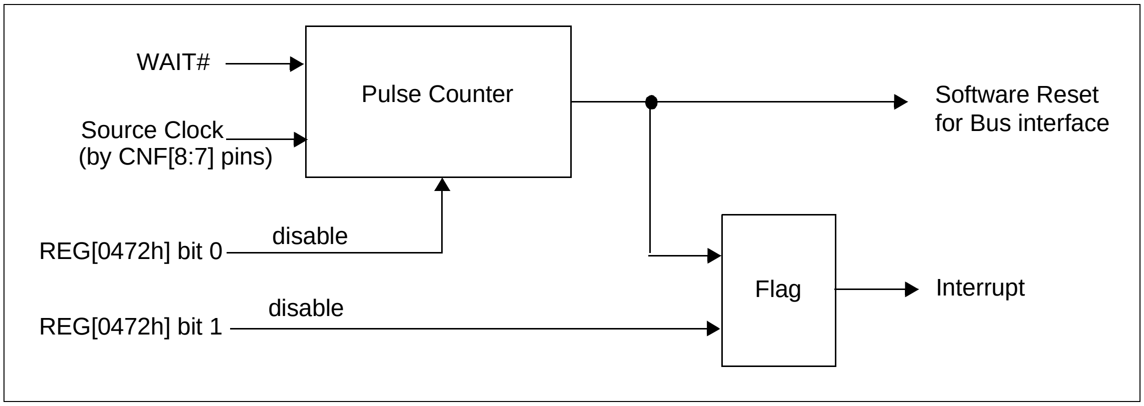


Figure 20-2: Host WAIT# Length Time-out Function

20.3 Indirect Interface

The S1D13513 supports Indirect host interfaces which use a different method of addressing the registers/memory. The following sections provide example sequences for each access type.

Table 20-2 : Indirect Interface Port

AB[2]	AB[1]	RD_x	WR_x	Register Name
0	0	0	1	Index Register
0	0	1	0	Index Register
0	1	0	1	Status Register
0	1	1	0	Reserved
1	0	0	1	Data register
1	0	1	0	Data register
1	1	0	1	Reserved
1	1	1	0	Reserved

Note

The name of RD_x, WR_x may be different, depending on the Host Bus type.

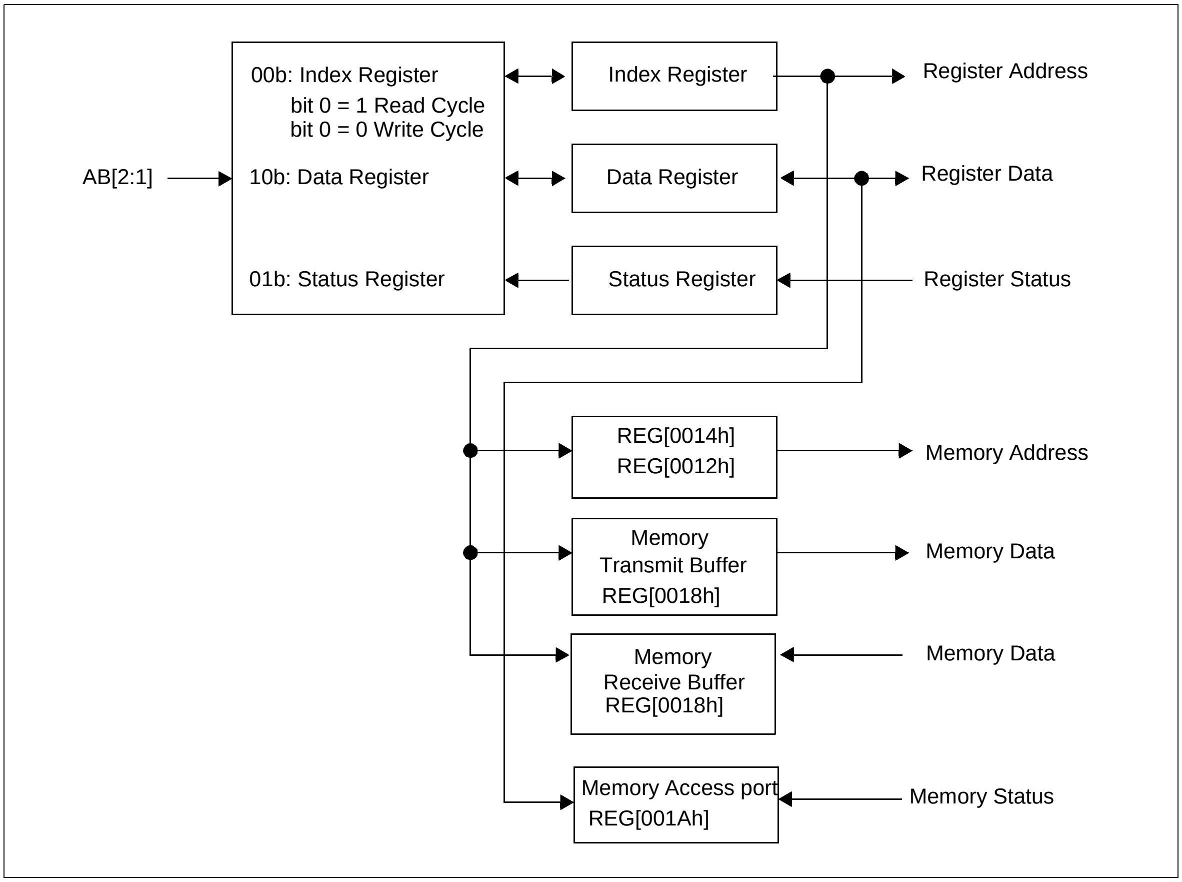


Figure 20-3: Indirect Interface Block Diagram

20.3.1 Indirect Addressing for Register Access

AB[2:1] = 00b Indirect Interface Index Register
Default = 0000h Read/Write

Register Address bits 15-8							
15	14	13	12	11	10	9	8
Register Address bits 7-1							Read/Write Cycle Select
7	6	5	4	3	2	1	0

bits 15-1 Register Address bits [15:1]
These bits are used for Parallel Indirect Interface modes only.
These bits set the register address for the indirect interface.

bit 0 Read/Write Cycle Select
This bit is used for Parallel Indirect Interface modes only.
This bit selects whether a read or a write is performed for the next data port access.
When this bit = 0, a write is performed.
When this bit = 1, a read is performed.

AB[2:1] = 01b Indirect Interface Status Register
Default = 0000h Read Only

n/a							
15	14	13	12	11	10	9	8
n/a							Memory Status
7	6	5	4	3	2	1	0

bit 0 Memory Status (Read Only)
This bit is used for Parallel Indirect Interface modes only.
This bit indicates the status of the Memory Controller. The status of this bit must be checked before accessing the memory, however confirmation for continuous memory accesses is not necessary.
When this bit = 0b, the memory controller is idle and the Host CPU can access memory.
When this bit = 1b, the memory controller is busy and the Host CPU cannot access memory.

AB[2:1] = 10b Indirect Interface Data Port Register
Default = 0000h Read/Write

Indirect Interface Data Port bits 15-8							
15	14	13	12	11	10	9	8
Indirect Interface Data Port bits 7-0							
7	6	5	4	3	2	1	0

bits 15-0 Indirect Interface Data Port bits [15:0]
These bits are used for Parallel Indirect Interface modes only.
These bits are used for read/write data transfers to the register address specified by bits 15-1 of the Indirect Interface Index Register, AB[2:1] = 00b.

20.3.2 Register Access

When the indirect host interface is selected, register accesses should follow the procedure below.

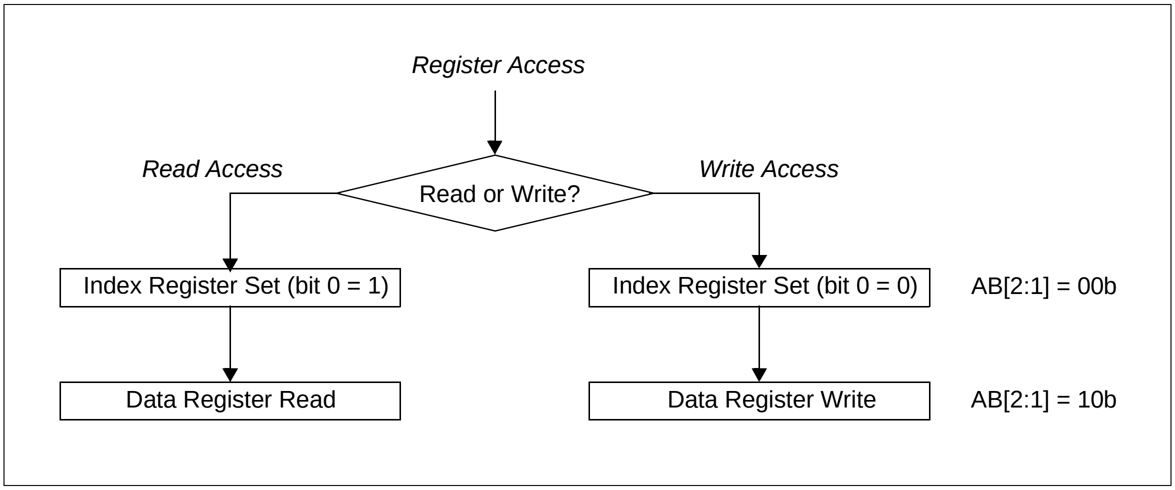


Figure 20-4: Register Access

20.3.3 Memory Access

When the indirect host interface is selected, memory accesses should follow the procedure below. When a memory read or write error occurs, re-start by setting the address again as the byte cannot be accessed.

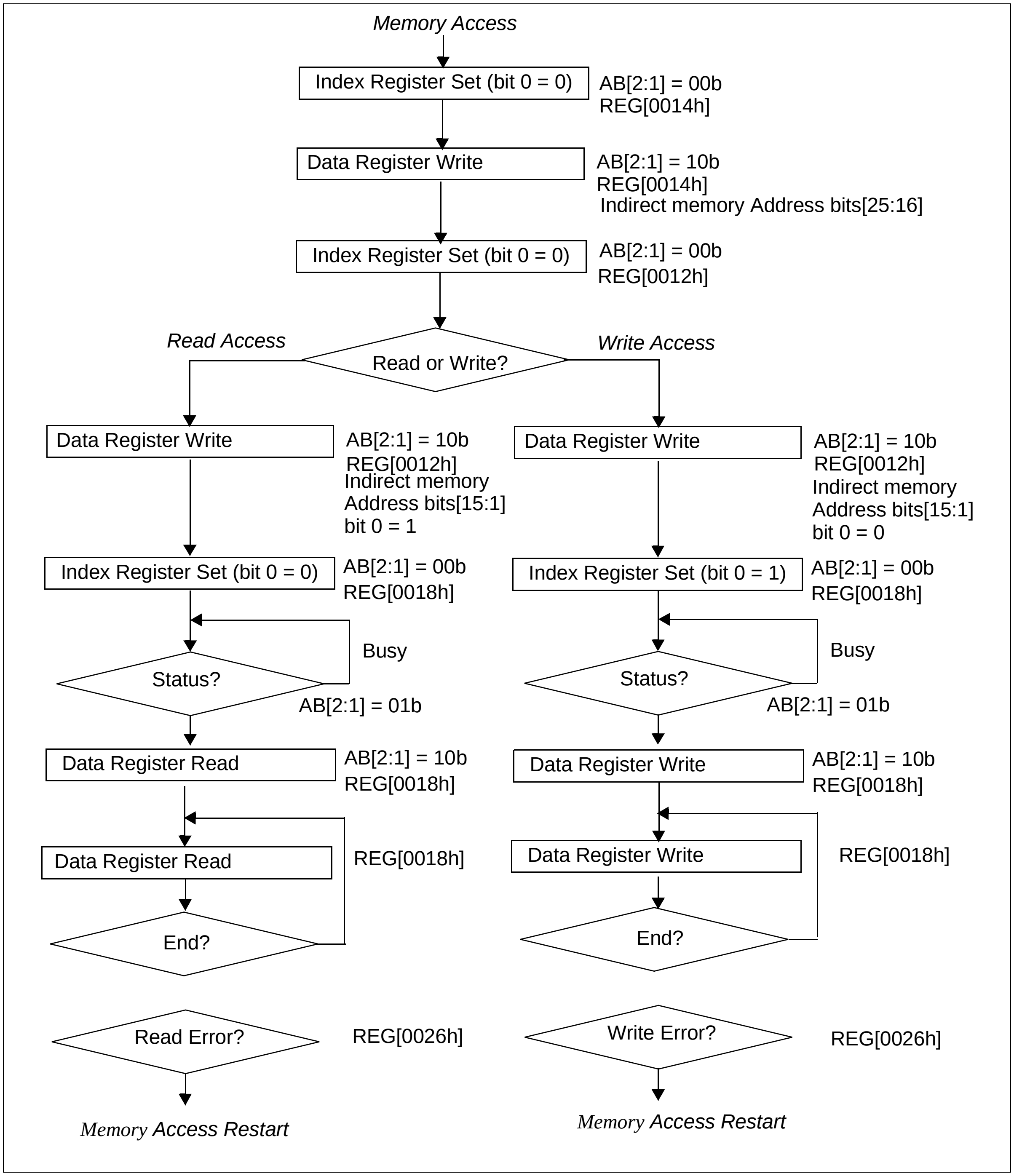


Figure 20-5: Memory Access

20.4 Read Ahead Feature

The Host interface includes a prefetch buffer which accelerates memory accesses by “reading ahead”. When a single read is issued from the Host interface, the S1D13513 internally reads memory using a burst transfer. The read data is saved in the prefetch buffer. If the next read address is a +2 address to the previous read address, the Host interface can return the data immediately without a new memory read access. This method provides much faster access as long as the memory address is incremented by 2 each time.

The prefetch buffer is cleared when the following occurs:

- a memory write access occurs
- the next read address is not a +2 address to the previous read address

Note

For S1D13513 revisions 00h or 01h (not revision 02h), a dummy write cycle is required between each read cycle except when incrementing the sequential addresses by 2 for each read. If the additional write cycle is not inserted, incorrect data may be read by the Host. The product revision can be checked by reading REG[0000h] bits 15-8.

Possible Issue with the Prefetch Buffer

In some particular use cases where one of the Camera, BitBLT, DMAC, or Sprite internal blocks write data to a memory address range AND the Host attempts to read data from the same memory address range, the Read Ahead feature may cause a coherency issue. The issue occurs because data already captured in the prefetch buffer may not be updated as the internal blocks write the data in the memory, resulting in the Host reading incorrect data.

If such a coherency issue occurs, the system must force the pre-buffered data to be discarded. The solution used depends on the type of Host interface: direct or indirect (parallel or serial).

Direct Host Interface

If the coherency issue occurs and the Direct Host interface is used, the pre-buffered data can be discarded by performing a dummy read or write to memory. Alternately, for S1D13513 revision 02h only, the prefetch buffer can be disabled by setting REG[0044h] bit 15 = 1b. However, disabling the prefetch buffer will decrease memory read performance.

Indirect Host Interface

If the coherency issue occurs and the Indirect Host interface (parallel or serial) is used, the pre-buffered data can be discarded by performing a dummy write from REG[0018h] (REG[0012h] bit 0 must equal 0b) after each read operation. Please note that for indirect interfaces, the prefetch buffer cannot be disabled.

Example of Coherency Issue

For example, assuming the Host tries to read data from memory address 0000h through 01FEhh and the internal hardware (Camera YRC, DMAC, BitBLT or Sprite) tries to write data to memory address 0200h through 03FEh, the issue occurs after the internal hardware transfer writes data to 0200h - 03FEh and the Host CPU tries to read data from memory address 0200h.

20.5 Serial Interface

The S1D13513 supports three types of Host CPU interface - parallel direct, parallel indirect, and serial indirect. The serial host interface uses a different method of addressing the registers/memory than the parallel interfaces. The following sections show example sequences for each access type.

20.5.1 Description

The S1D13513 Serial Host Interface supports the following.

- 16-bit transfers
- maximum frequency of 16MHz
- write and read data transfers
    - single transfers
    - burst transfers for the Memory
- reading the indirect interface status
- sending the MSB first
- selectable serial host connections on HVDD1 or HVDD2

Five signals are used for the serial interface.

- SI: Serial Data In
- SO: Serial Data Out
- SCK: Serial Clock
- CS#/SCS#: Serial Chip Select
- SA0: Command / Data Select (0 = command, 1 = data)

The SA0 Command / Data pin is used to distinguish between data transfers and command transfers. Command transfers are used to initiate either a Read, Write, or Get Status transfer as follows.

Table 20-3: Serial Host Interface Commands

Command	Value	Function
CMD_WRITE	0000h	Start Write Cycle
CMD_READ	4000h	Start Read Cycle
CMD_END	8000h	End Cycle
CMD_STATUS	C000h	Read Indirect Interface Status

The Serial Host Interface uses the Indirect Data Bus in the S1D13513 to set the register address and read/write data. The Serial Host Interface has access to the Indirect Interface Status Register which can be read using the CMD_STATUS command. This returns the Memory Status.

The Serial Data Out pin (SO), is Hi-Z when CS# is high and is active when CS# is low.

20.5.2 Burst Mode Operation

Serial host interface burst mode allows the CS# line to be held low and then continuously send or receive data to/from the S1D13513.

CS# can either be held low between each 16-bit transfer or it can be toggled, but it must change from high to low before the Command portion of the transfer.

The following figure shows burst mode when CS# is held low.

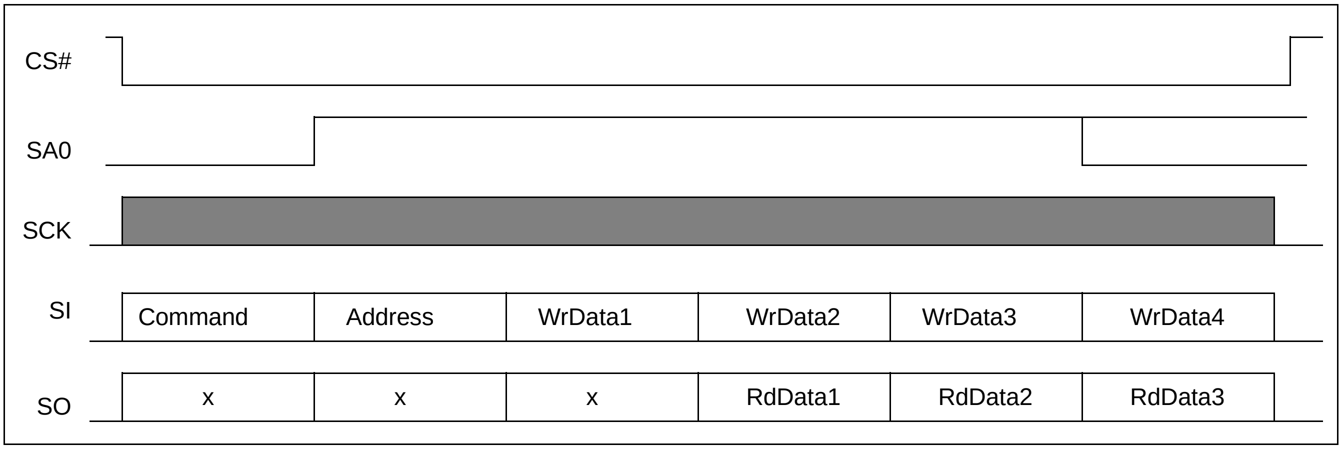


Figure 20-6: Serial Host Interface Burst Mode for CS# Held Low

The following figure shows burst mode when CS# is toggling.

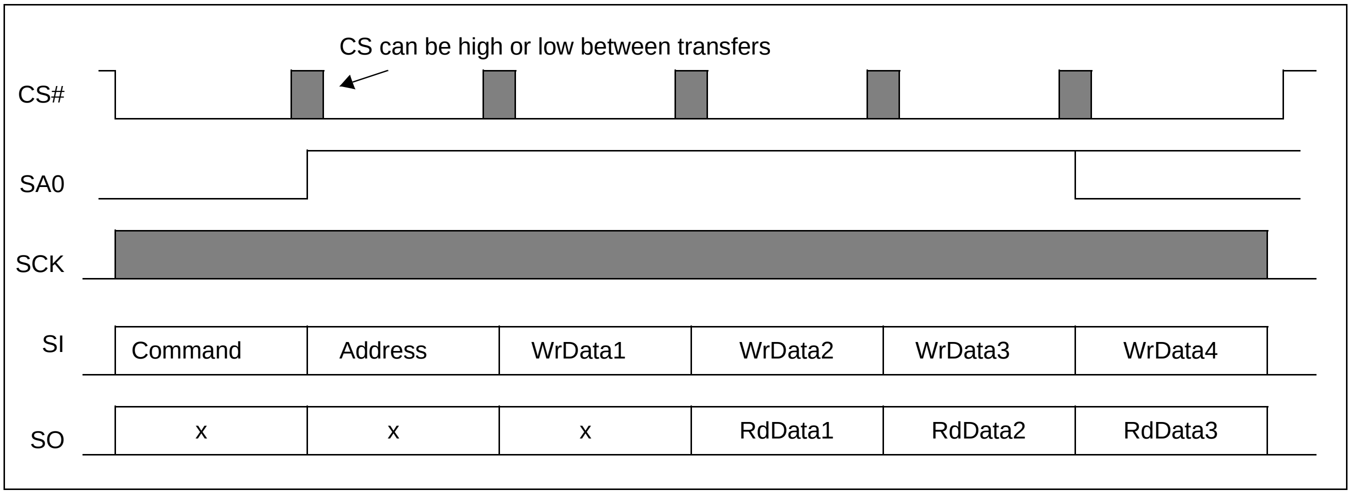


Figure 20-7: Serial Host Interface Burst Mode for CS# Toggling

Note

If CS# goes high between transfers, then it must be held high for at least one SCK period between the last bit and first bit.

20.5.3 Data Transfer Examples

In the following examples X = unknown data

Writing Single Data

1. Send the Write Command.
2. Send the register address where data is to be written.
3. Send the data to be written.

Table 20-4: Writing Single Data

SA0	Data Sent	Data Received
0	CMD_WRITE	X
1	Register address where data is to be written	X
1	Data to be written	X
0	CMD_END	

Writing Burst Data

1. Send the Write Command.
2. Send the register address where data is to be written.
3. Send the data to be written.
4. Repeat step 3 until all data has been written.

Table 20-5: Writing Burst Data

SA0	Data Sent	Data Received
0	CMD_WRITE	X
1	Register address where data is to be written	X
1	Data to be written	X
1	Data to be written	X
...	...	...
1	Data to be written	X
0	CMD_END	

Reading Single Data

1. Send the Read Command.
2. Send the register address that data is to be read from.
3. Write a value. This performs the read operation and captures the data from the specified address.
4. Send the Read End command. This returns the data previously captured and does not perform another read from the register address.

Table 20-6: Reading Single Data

SA0	Data Sent	Data Received
0	CMD_READ	X
1	Register address to read data from	X
1	0	X
0	CMD_END	Read Data from previous read

Reading Burst Data

1. Send the Read Command.
2. Send the register address that data is to be read from.
3. Write a value. This performs the read operation and captures the data from the specified address.
4. Write another value. This returns the data captured from the previous read and performs another read at the same address which captures the data.
5. Repeat step 4 until all the data minus 1 has been read.
6. Send the Read End command. This returns the data previously captured and does not perform another read from the register address.

Table 20-7: Reading Burst Data

SA0	Data Sent	Data Received
0	CMD_READ	X
1	Register address to read data from	X
1	0	X
1	0	Read Data from previous read
1	...	Read Data from previous read
...	...	...
0	CMD_END	Read Data from previous read

Reading the Indirect Interface Status

1. Send the Status Command.
2. Send the Read End command. This returns the Indirect Status.

Table 20-8: Reading the Indirect Interface Status

SA0	Data Sent	Data Received
0	CMD_STATUS	X
0	CMD_END	Indirect Status

Note

For details on the information returned in the Indirect Interface Status register, see Section 20.3.1, “Indirect Addressing for Register Access” on page 439.

20.5.4 Indirect Register Address Auto Increment Serial Examples

Write 8 words to LUT1 starting at address 400h.

Host Bus Cycle	S1D13513 Operation
Send Command Write	
Set Address to 400h	Set Indirect Address to 0400h
Send Data = 1234h	Write 1234h to Address 0400h
Send Data = 0056h	Write 0056h to Address 0402h
Send Data = 789Ah	Write 789Ah to Address 0404h
Send Data = 00BCh	Write 00BCh to Address 0406h

Write 2 words to address 210h and 212h and then 2 words to address 218h and 21Ah.

Host Bus Cycle	S1D13513 Operation
Send Command Write	
Set Address to 0210h	Set Indirect Address to 0210h
Send Data = 1234h	Write 1234h to Address 0210h
Send Data = 5678h	Write 5678h to Address 0212h
Send Command Write	
Set Address to 0218h	Set Indirect Address to 0218h
Send Data = 1234h	Write 1234h to Address 0218h
Send Data = 5678h	Write 5678h to Address 021Ah

Read 3 words from address 220h to 224h.

Host Bus Cycle	S1D13513 Operation
Send Command Read	
Set Address to 0220h	Set Indirect Address to 0220h
Read Dummy Data	Read from Address 0220h
Read Data from address 220h	Read from Address 0222h
Read Data from address 222h	Read from Address 0224h
Send Command Read End and Read	
Data from address 224h	

20.5.5 Serial Host Voltage Selection

The Host processor may be connected to the Host pins on HVDD1 or the FPDAT pins on HVDD2 based on the settings of CNF[4:0]. CNF[4:0] = 10000b and 11000b selects Serial Host connections on HVDD1. CNF[4:0] = 10001b and 11001b selects Serial Host connections on HVDD2. The CNF settings must be selected before power on as the settings are latched on the rising edge of RESET#.

Note

20.6 Register Accesses for Big Endian Host Interfaces

When a Big Endian host interface is selected (see Section 5.3, “Summary of Configuration Options” on page 41), the registers must be accessed using a method which “byte-swaps” the upper and lower data byte in each register. This requirement applies for both read and write accesses.

For example, reading the S1D13513 Product Code from a Little Endian host will return 00h from the upper byte (bits 15-8) and 2Ch from the lower byte (bits 7-0). This results in a product code of 002Ch which is exactly as described for REG[0002h] in Section 10.4.1, “Host Interface Registers” on page 133. If no adjustments are made, a Big Endian host will read the same register and return a product code of 2Ch from the upper byte and 00h from the lower byte. This would result in a product code of 2C00h which does not identify the S1D13513 correctly. Register writes must also be adjusted or the S1D13513 will be configured incorrectly.

The following code provides an example of how the required byte-swap can be performed.

```
#define BIG_ENDIAN
#undef  LITTLE_ENDIAN

#if defined(LITTLE_ENDIAN) && !defined(BIG_ENDIAN)

     #define BYTE_SWAP(x) (x)

#elif defined(BIG_ENDIAN) && !defined(LITTLE_ENDIAN)

     #define BYTE_SWAP(x) (((UInt16)(x) & 0xFF) << 8) | (((UInt16)(x) & 0xFF00) >> 8 ))

#else

#error "Please define either BIG_ENDIAN or LITTLE_ENDIAN."

#endif


UInt16 seReadReg16( UInt32 Index )
{
     UInt16 val =  *(UInt16*)(gRegisterAddress + Index);
     return BYTE_SWAP(val);
}

void seWriteReg16( UInt32 Index, UInt16 Value )
{
     *(UInt16*)(gRegisterAddress + Index) = (UInt16)(BYTE_SWAP(Value));
}
```

21 LCD Panel Interface

The S1D13513 stores image data in external SDRAM memory for output to a single panel including the following LCD panel types.

Note
24-bit panels are not supported for the QFP package.

- 16/18/24-bit TFT/ND-TFD (includes serial command interface support)
  - ND-TFD 4-pin interface (8-bit)
  - ND-TFD 3-pin interface (9-bit)
  - µ-Wire TFT interface (16-bit)
  - 24-bit Serial interface
  - Serial command interface has configurable bit direction, phase, and polarity
- 16/18/24-bit HR-TFT
- 8-bit Monochrome or Color Type 2 Passive Panels
- YUV Digital Output
  - YUV 4:2:2 can be output to support TV via an external video encoder
- For information on supported resolutions, see Section 18.2, “Memory Bandwidth” on page 428

Note
Passive panels have maximum display size limitations. Please contact your EPSON representative for details.

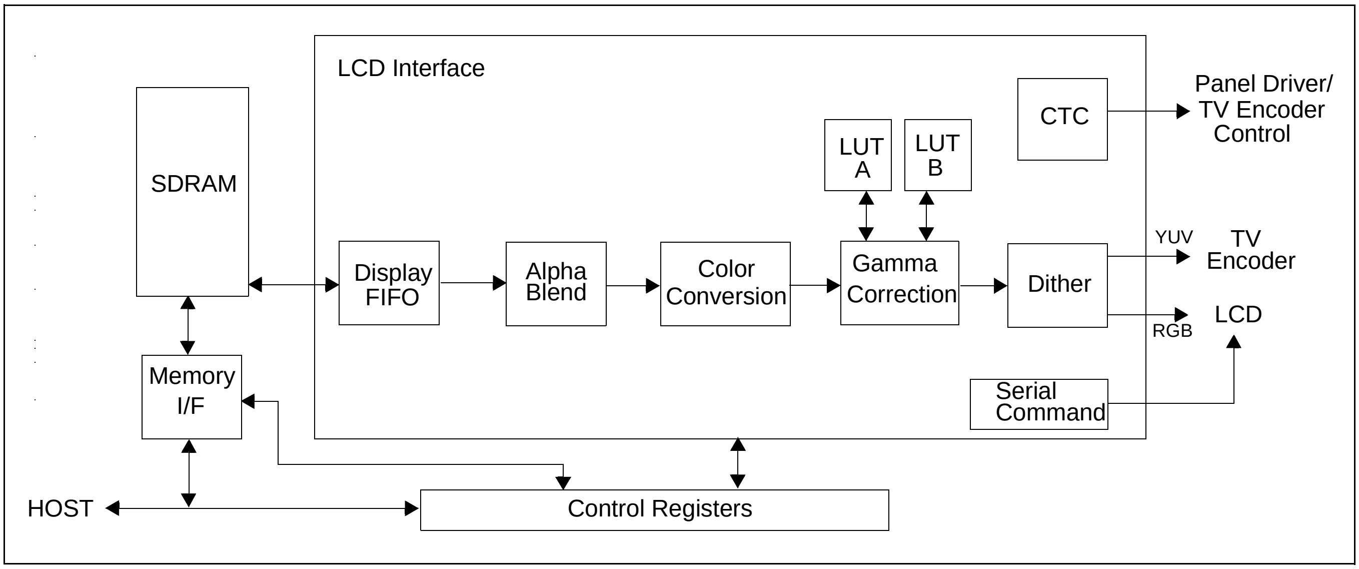


Figure 21-1: LCD Interface Overview

21.1 TFT/ND-TFD Panels

A TFT/ND-TFD RGB interface panel is selected when REG[0800h] bit 14 = 0b and REG[0800h] bits 2-0 = 000b. Data bus widths of 16, 18, and 24-bits are supported and selected using REG[0800h] bits 10-8. For some TFT panel types a Serial Command Interface is used to issue command/parameter information to the panel. The type of Serial Command Interface is selected using REG[0816h] bits 7-5.

For pin mapping details, refer to Section 5.5, “LCD Interface Pin Mapping” on page 48.

21.1.1 TFT/ND-TFD Data Output Formats

The following information shows the data output formats for each panel data bus width.

- 16-bit RGB interface LCD panel RGB 5:6:5 (REG[0800h] bits 10-8 = 001b)
- 18-bit RGB interface LCD panel RGB 6:6:6 (REG[0800h] bits 10-8 = 010b)
- 24-bit RGB interface LCD panel RGB 8:8:8 (REG[0800h] bits 10-8 = 011b)

Table 21-1: 16/18/24-Bit TFT/ND-TFD Data Output Formats

S1D13513 Pin	16-Bit	18-Bit	24-Bit
FPDAT0	R^4	R^5	R^7
FPDAT1	R^3	R^4	R^6
FPDAT2	R^2	R^3	R^5
FPDAT3	G^5	G^5	G^7
FPDAT4	G^4	G^4	G^6
FPDAT5	G^3	G^3	G^5
FPDAT6	B^4	B^5	B^7
FPDAT7	B^3	B^4	B^6
FPDAT8	B^2	B^3	B^5
FPDAT9	R^1	R^2	R^4
FPDAT10	R^0	R^1	R^3
FPDAT11	Low	R^0	R^2
FPDAT12	G^2	G^2	G^4
FPDAT13	G^1	G^1	G^3
FPDAT14	G^0	G^0	G^2
FPDAT15	B^1	B^2	B^4
FPDAT16	B^0	B^1	B^3
FPDAT17	Low	B^0	B^2
FPDAT18	Low	Low	R^1
FPDAT19	Low	Low	R^0
FPDAT20	Low	Low	G^1
FPDAT21	Low	Low	G^0
FPDAT22	Low	Low	B^1
FPDAT23	Low	Low	B^0

21.1.2 RGB Serial Command Interfaces

When a RGB interface panel with a serial command interface is selected (see REG[0816h] bits 7-5, the GPIOG[3:0] pins are used according to the serial command interface type.

- General TFT panel (REG[0816h] bits 7-5 = XXXb)
- ND-TFD 4-pin panel (REG[0816h] bits 7-5 = 000b)
- ND-TFD 3-pin panel (REG[0816h] bits 7-5 = 001b)
- a-Si TFT panel (REG[0816h] bits 7-5 = 010b)
- μ-Wire TFT panel (REG[0816h] bits 7-5 = 100b)
- 24-bit serial data RGB interface LCD panel (REG[0816h] bits 7-5 = 101b)

Table 21-2: RGB Serial Command Interfaces Pin Usage Summary

Interface Type	**S1D13513 Pin**			
	GPIOG0	**GPIOG1**	**GPIOG2**	**GPIOG3**
General TFT	—	—	—	—
ND-TFD 4-pin	XCS	SCK	A0	SO
ND-TFD 3-pin	XCS	SCK	—	SO
a-Si TFT	SSTB	SCLK	—	SDATA
uWIRE TFT	LCDCS	SCLK	—	SDO
24-bit Serial TFT	XCS	SCK	—	SO

For timing details, refer to Section 7.6, “Panel Interface Timing” on page 90.

21.1.3 TFT/ND-TFD Programming Flow

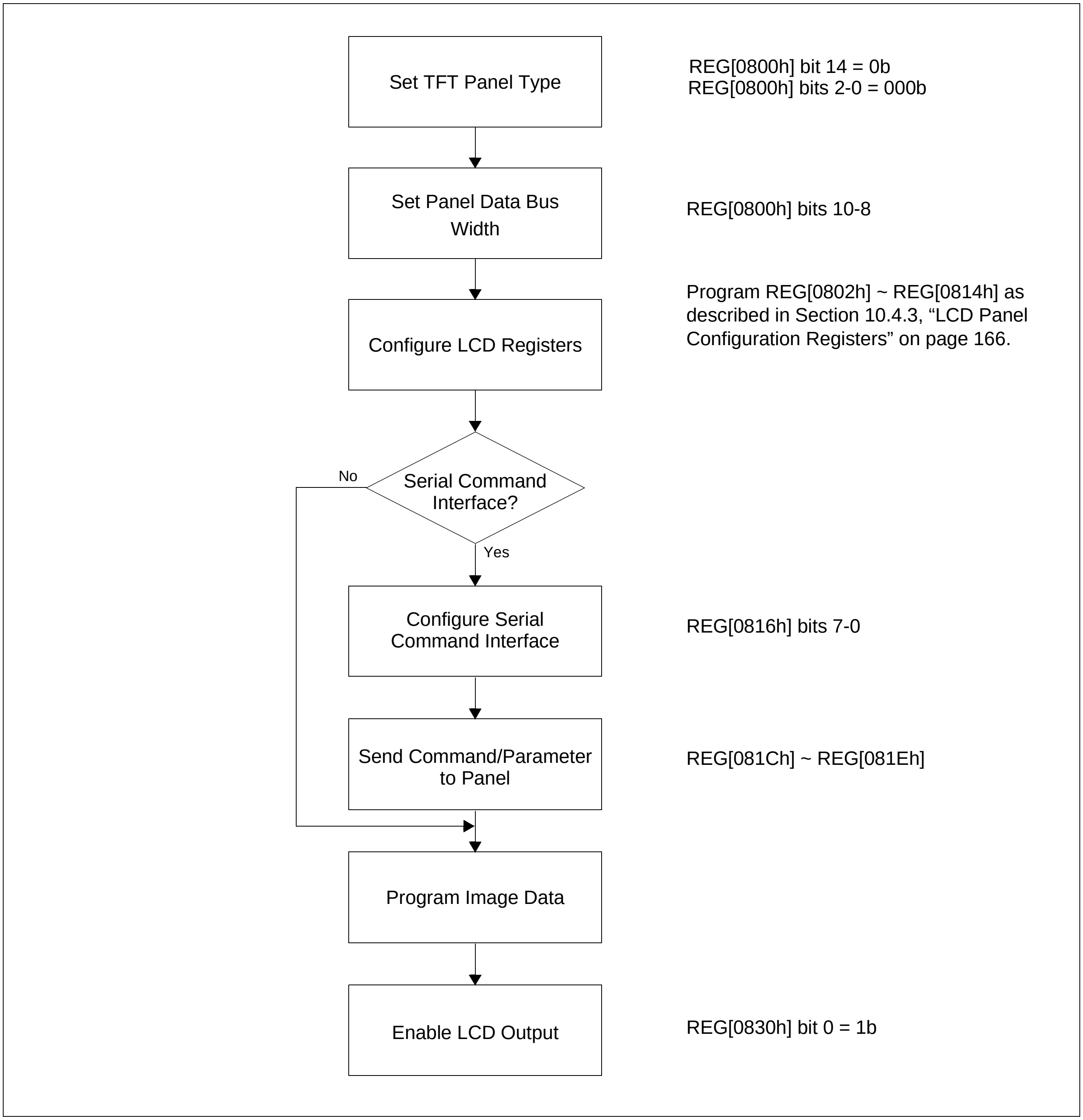

Set TFT Panel Type
REG[0800h] bit 14 = 0b
REG[0800h] bits 2-0 = 000b
Set Panel Data Bus Width
REG[0800h] bits 10-8
Configure LCD Registers
Program REG[0802h] ~ REG[0814h] as described in Section 10.4.3, “LCD Panel Configuration Registers” on page 166.
Serial Command Interface?
No
Yes
Configure Serial Command Interface
REG[0816h] bits 7-0
Send Command/Parameter to Panel
REG[081Ch] ~ REG[081Eh]
Program Image Data
Enable LCD Output
REG[0830h] bit 0 = 1b

Figure 21-2: TFT/ND-TFD Programming Flow

21.2 HR-TFT Panels

A HR-TFT RGB interface panel is selected when REG[0800h] bit 14 = 0b and REG[0800h] bits 2-0 = 010b. Data bus widths of 16, 18, and 24-bits are supported and selected using REG[0800h] bits 10-8. HR-TFT panels use additional pins compared to a standard TFT interface (see Section 21.2.2, “HR-TFT Interface Pins” on page 458).

For pin mapping details, refer to Section 5.5, “LCD Interface Pin Mapping” on page 48.

21.2.1 HR-TFT Data Output Formats

The following information shows the data output formats for each panel data bus width.

- 16-bit RGB interface LCD panel RGB 5:6:5 (REG[0800h] bits 10-8 = 001b)
- 18-bit RGB interface LCD panel RGB 6:6:6 (REG[0800h] bits 10-8 = 010b)
- 24-bit RGB interface LCD panel RGB 8:8:8 (REG[0800h] bits 10-8 = 011b)

Table 21-3: 16/18/24-Bit HR-TFT Data Output Formats

Pin	16-Bit	18-Bit	24-Bit
FPDAT0	R^4	R^5	R^7
FPDAT1	R^3	R^4	R^6
FPDAT2	R^2	R^3	R^5
FPDAT3	G^5	G^5	G^7
FPDAT4	G^4	G^4	G^6
FPDAT5	G^3	G^3	G^5
FPDAT6	B^4	B^5	B^7
FPDAT7	B^3	B^4	B^6
FPDAT8	B^2	B^3	B^5
FPDAT9	R^1	R^2	R^4
FPDAT10	R^0	R^1	R^3
FPDAT11	Low	R^0	R^2
FPDAT12	G^2	G^2	G^4
FPDAT13	G^1	G^1	G^3
FPDAT14	G^0	G^0	G^2
FPDAT15	B^1	B^2	B^4
FPDAT16	B^0	B^1	B^3
FPDAT17	Low	B^0	B^2
FPDAT18	Low	Low	R^1
FPDAT19	Low	Low	R^0
FPDAT20	Low	Low	G^1
FPDAT21	Low	Low	G^0
FPDAT22	Low	Low	B^1
FPDAT23	Low	Low	B^0

21.2.2 HR-TFT Interface Pins

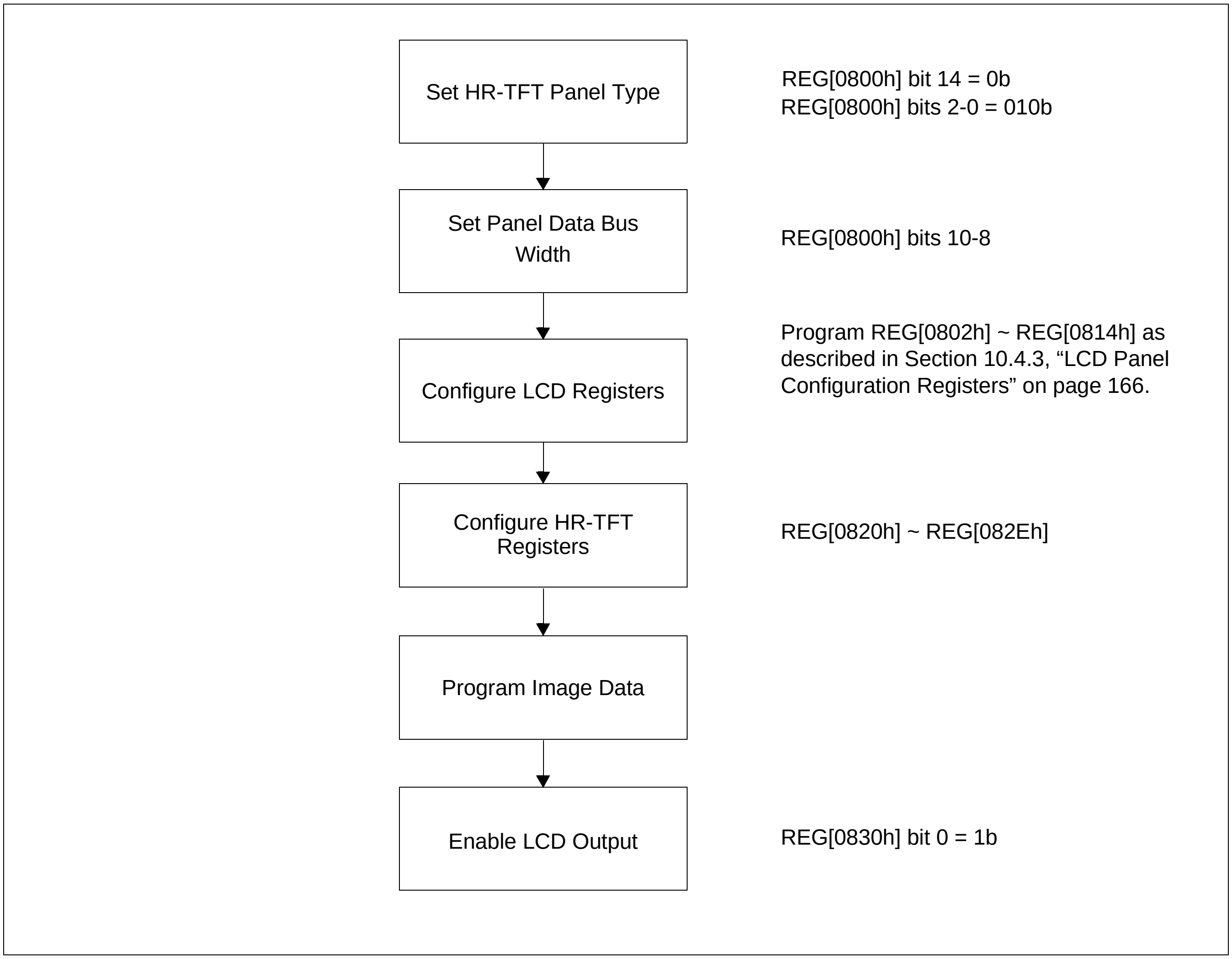
When a HR-TFT panel is selected (REG[0800h] bits 2-0 = 010b), the GPIOG[4:0] pins are used to provide the following signals.

Table 21-4: HR-TFT Interfaces Summary

Interface Type	GPIOG0	GPIOG1	GPIOG2	GPIOG3	GPIOG4
HR-TFT	PS	CLS	REV	SPL	SPR

For timing details for the HR-TFT interface, refer to Section 7.6.2, “HR-TFT Panel Timing” on page 93.

21.2.3 HR-TFT Programming Flow



REG[0800h] bit 14 = 0b
REG[0800h] bits 2-0 = 010b

REG[0800h] bits 10-8

Program REG[0802h] ~ REG[0814h] as described in Section 10.4.3, “LCD Panel Configuration Registers” on page 166.

REG[0820h] ~ REG[082Eh]

REG[0830h] bit 0 = 1b

Figure 21-3: HR-TFT Programming Flow

21.3 Passive Panels

A Passive interface panel is selected when REG[0800h] bit 14 = 1b. The passive panel type can be either Single Monochrome or Single Color Format Type 2 (REG[0800h] bits 13-11). Both passive panel types use an 8-bit data bus so REG[0800h] bits 10-8 must be set to 001b.

For pin mapping details, refer to Section 5.5, “LCD Interface Pin Mapping” on page 48.

21.3.1 Passive Panel Data Output Format

The following information shows the data output format for passive panels.

- Single Monochrome (REG[0800h] bits 13-11 = 000b)
- Single Color Format Type 2 (REG[0800h] bits 13-11 = 010b)

Table 21-5: Passive Panel Data Output Formats

S1D13513 Pin	Single Monochrome	Single Color Type 2
FPDAT0	D0	D0
FPDAT1	D1	D1
FPDAT2	D2	D2
FPDAT3	D3	D3
FPDAT4	D4	D4
FPDAT5	D5	D5
FPDAT6	D6	D6
FPDAT7	D7	D7
FPDAT8	Driven 0	Driven 0
FPDAT9	Driven 0	Driven 0
FPDAT10	Driven 0	Driven 0
FPDAT11	Driven 0	Driven 0
FPDAT12	Driven 0	Driven 0
FPDAT13	Driven 0	Driven 0
FPDAT14	Driven 0	Driven 0
FPDAT15	Driven 0	Driven 0

21.3.2 Passive Panel Programming Flow

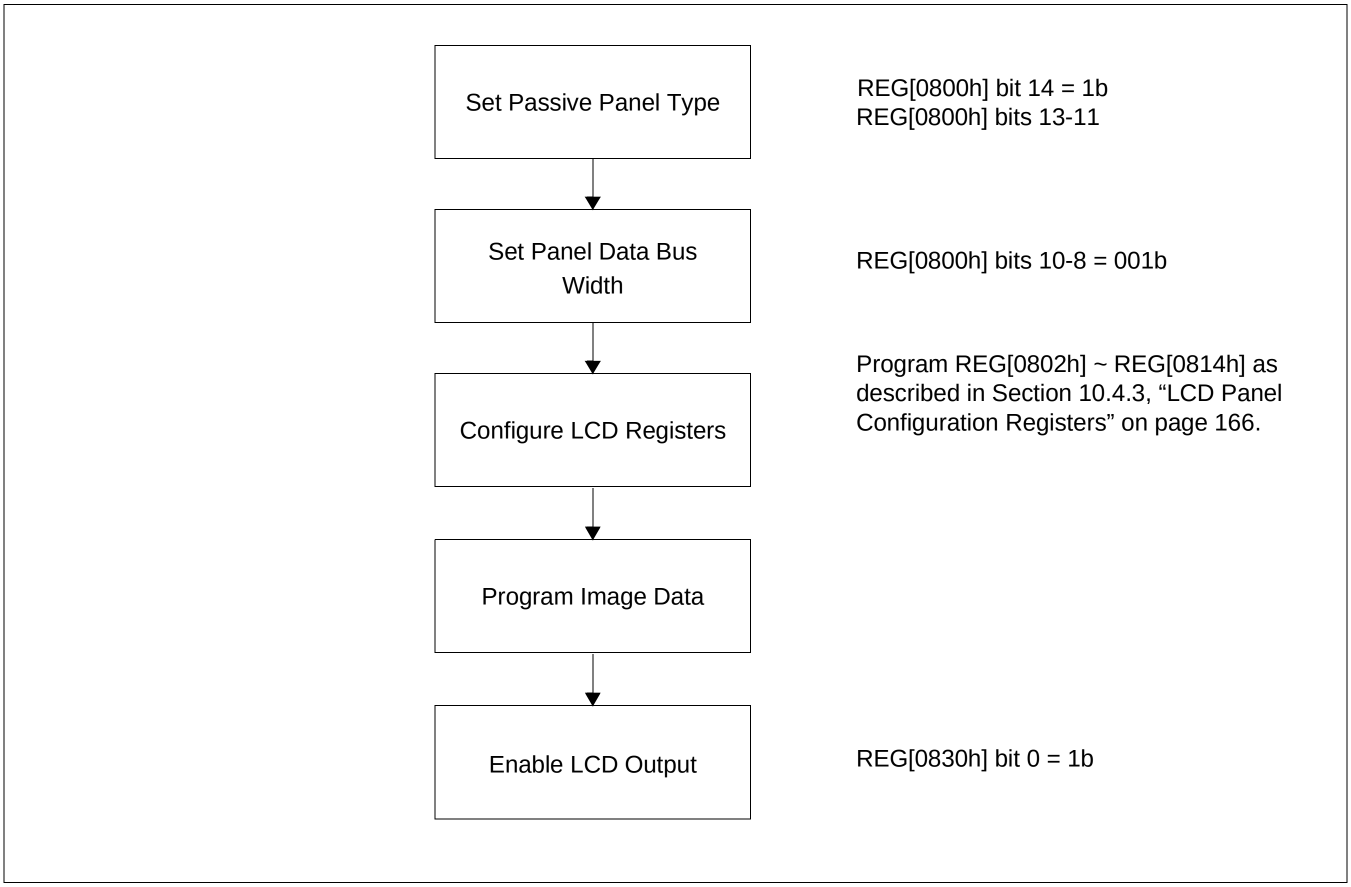


Figure 21-4: Passive Panel Programming Flow

21.4 YUV Digital Output

YUV Digital Output allows YUV 4:2:2 data to be output on the GPIOC[7:0] pins when REG[0800h] bits 2-0 = 100b.

For details on pin mapping, see Section 5.7, “YUV Digital Output Interface Pin Mapping” on page 51. For details on AC timing, see Section 7.6.8, “YUV Digital Output” on page 101.

21.4.1 YUV Digital Data Output Format

The following information shows the data output format for YUV Digital Output.

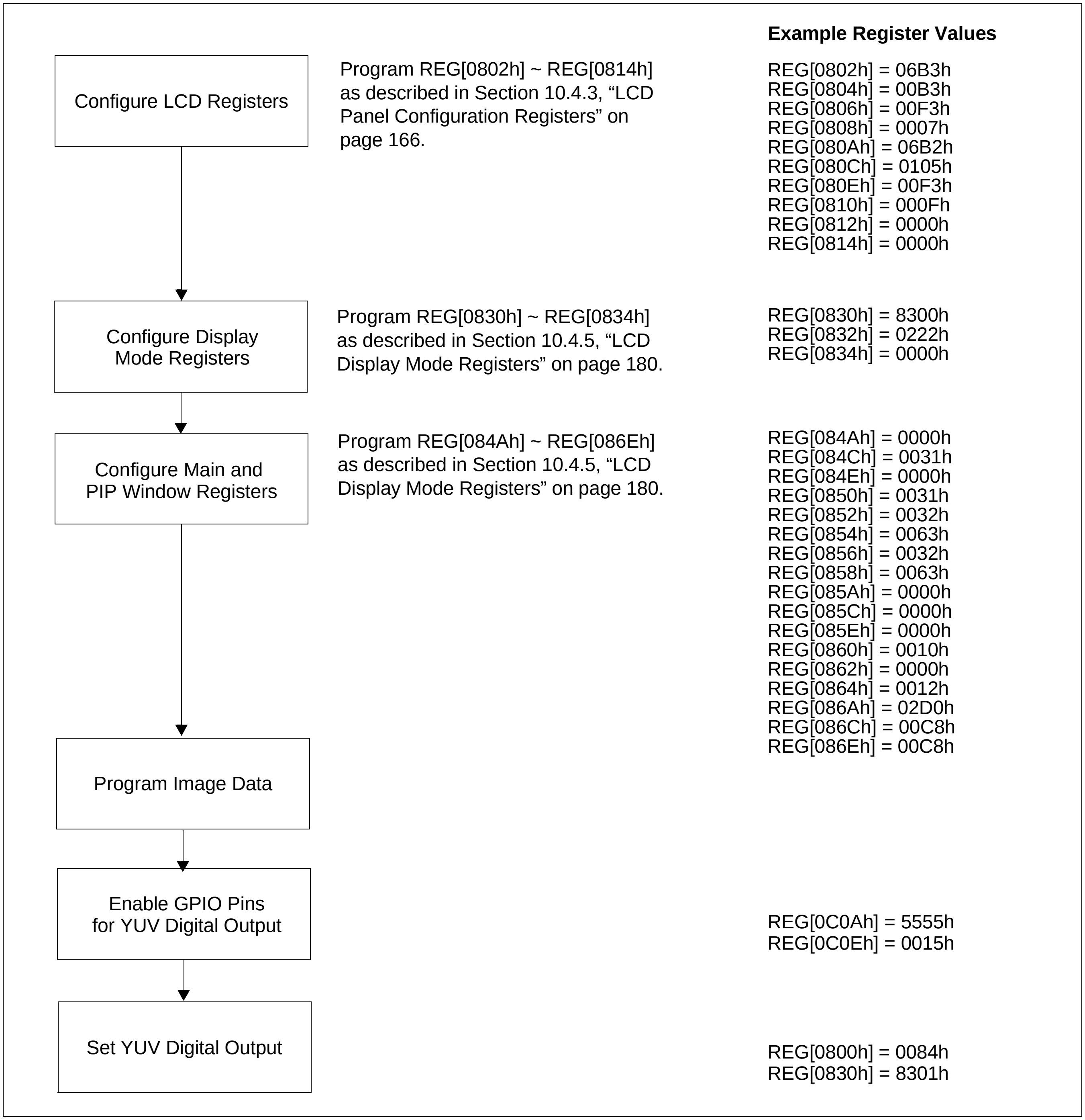
Table 21-6: YUV Digital Data Output Format

Cycle Count	1	2	3	4	5	...
D7	U_0^7	Y_0^7	V_0^7	Y_1^7	U_2^7	...
D6	U_0^6	Y_0^7	V_0^7	Y_1^7	U_2^7	...
D5	U_0^5	Y_0^7	V_0^7	Y_1^7	U_2^7	...
D4	U_0^4	Y_0^7	V_0^7	Y_1^7	U_2^7	...
D3	U_0^3	Y_0^7	V_0^7	Y_1^7	U_2^7	...
D2	U_0^2	Y_0^7	V_0^7	Y_1^7	U_2^7	...
D1	U_0^1	Y_0^7	V_0^7	Y_1^7	U_2^7	...
D0	U_0^0	Y_0^7	V_0^7	Y_1^7	U_2^7	...

21.4.2 YUV Digital Output Programming Flow

The following flow diagram shows how to setup YUV Digital Output for the following example case. Note that the display output is doubled when output to the TV encoder.

Display: 360x242x16 bpp
Main Window: 360x242x16 bpp
PIP1 Window: 100x100x16 bpp
PIP2 Window: 100x100x16 bpp



		Example Register Values

Configure LCD Registers	Program REG[0802h] ~ REG[0814h] as described in Section 10.4.3, “LCD Panel Configuration Registers” on page 166.	REG[0802h] = 06B3h
		REG[0804h] = 00B3h
		REG[0806h] = 00F3h
		REG[0808h] = 0007h
		REG[080Ah] = 06B2h
		REG[080Ch] = 0105h
		REG[080Eh] = 00F3h
		REG[0810h] = 000Fh
		REG[0812h] = 0000h
		REG[0814h] = 0000h

Configure Display Mode Registers	Program REG[0830h] ~ REG[0834h] as described in Section 10.4.5, “LCD Display Mode Registers” on page 180.	REG[0830h] = 8300h
		REG[0832h] = 0222h
		REG[0834h] = 0000h

Configure Main and PIP Window Registers	Program REG[084Ah] ~ REG[086Eh] as described in Section 10.4.5, “LCD Display Mode Registers” on page 180.	REG[084Ah] = 0000h
		REG[084Ch] = 0031h
		REG[084Eh] = 0000h
		REG[0850h] = 0031h
		REG[0852h] = 0032h
		REG[0854h] = 0063h
		REG[0856h] = 0032h
		REG[0858h] = 0063h
		REG[085Ah] = 0000h
		REG[085Ch] = 0000h
		REG[085Eh] = 0000h
		REG[0860h] = 0010h
		REG[0862h] = 0000h
		REG[0864h] = 0012h
		REG[086Ah] = 02D0h
		REG[086Ch] = 00C8h
		REG[086Eh] = 00C8h

Program Image Data		

Enable GPIO Pins for YUV Digital Output		REG[0C0Ah] = 5555h
		REG[0C0Eh] = 0015h

Set YUV Digital Output		REG[0800h] = 0084h
		REG[0830h] = 8301h

Figure 21-5: YUV Digital Output Programming Flow

22 Camera Interface

The S1D13513 is designed with a 2-port Camera interface that supports camera modules up to a maximum size of 640x480 (VGA) at 15 fps. The camera interface has an 8/16-bit data bus and can be configured for two 8-bit input ports or one 16-bit input port. Note that when configured for two 8-bit ports, only one camera can be used at a time.

Note
The Camera2 interface is not available for the QFP package.

The camera interface is designed to receive YUV 4:2:2 format image data synchronized with the camera clocks. Once the data is received from the camera module, it can be modified using the View or Capture resizer (see Section 14, “Resizers” on page 393) before it is output to the YUV to RGB Converter (YRC). The YRC can convert YUV data to RGB data for display, or can be bypassed allowing YUV data to be stored directly to the SDRAM. Alternately, it can be configured to receive raw JPEG data from JPEG capable camera modules.

The following figure provides an overview of the Camera Interface.

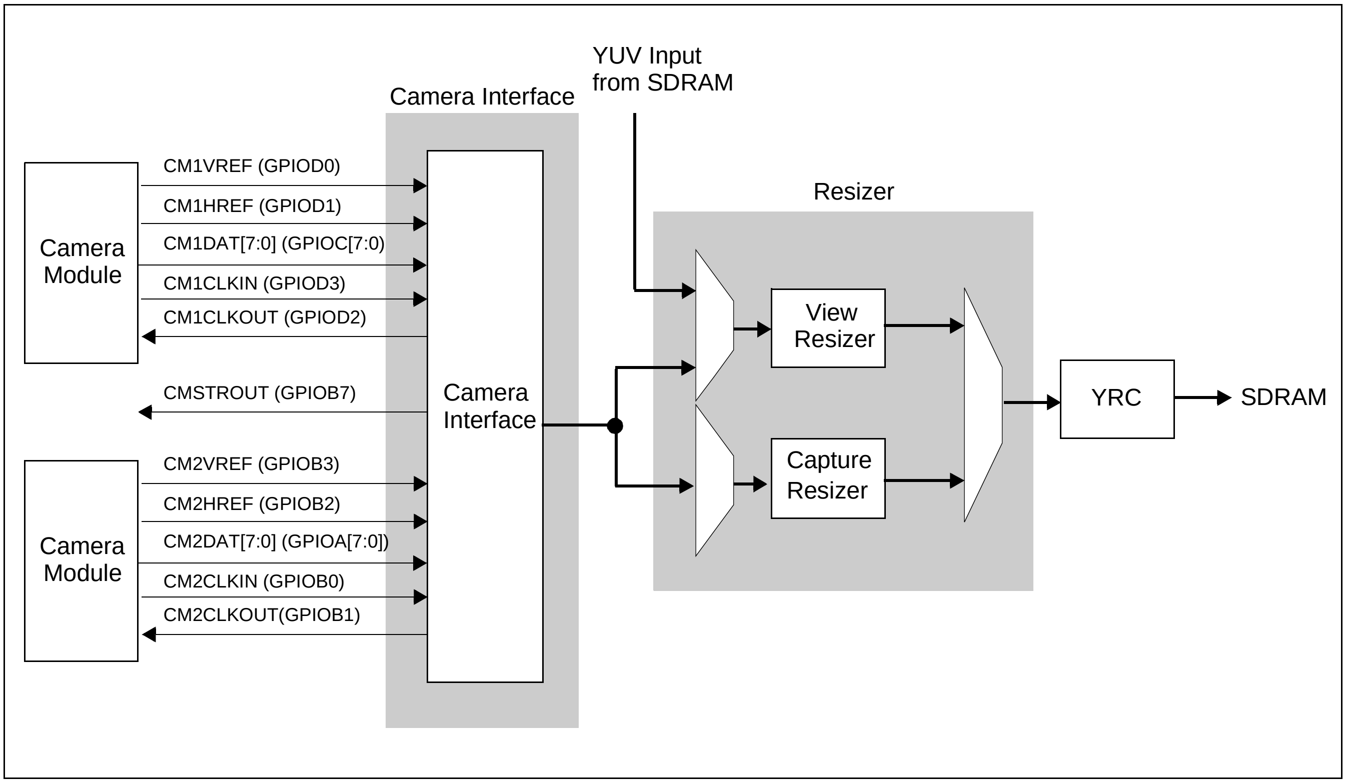


Figure 22-1: Camera Interface Overview (8-bit Data Bus Mode)

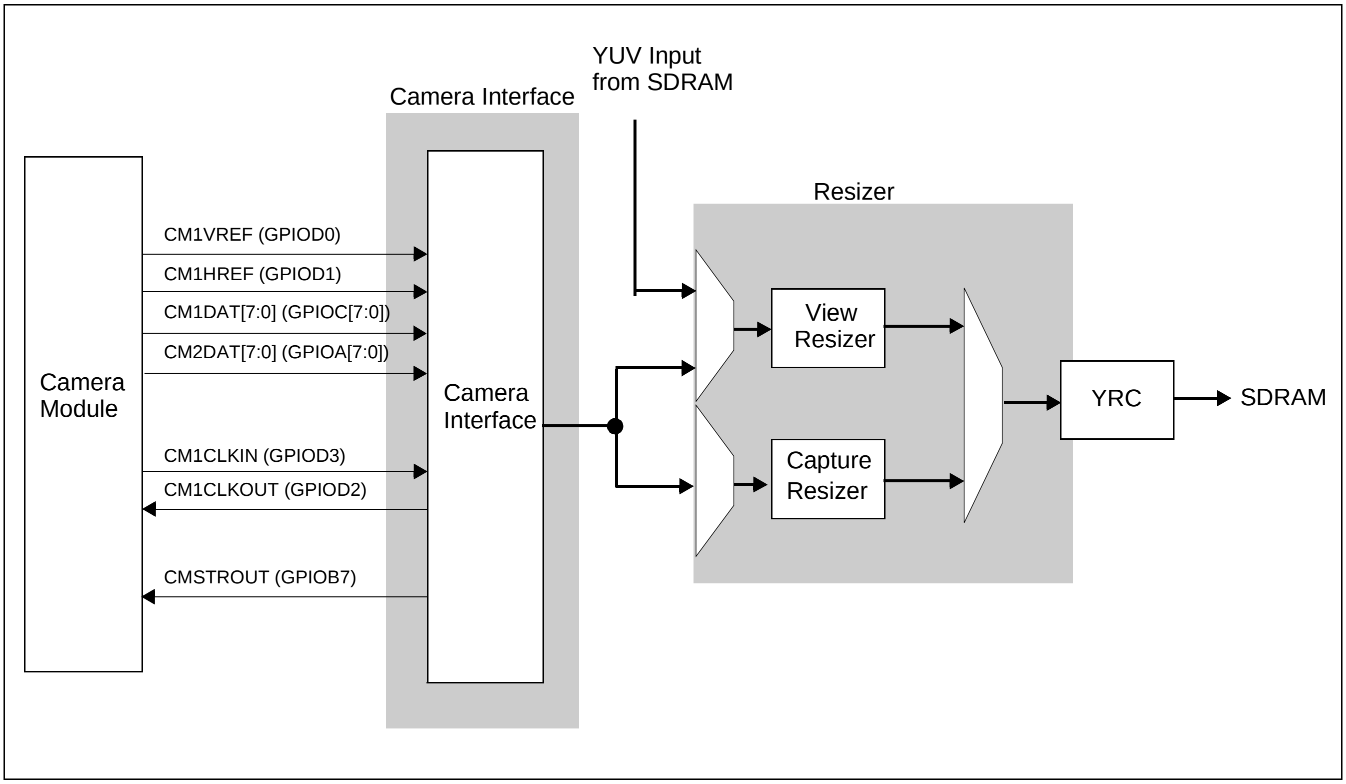


Figure 22-2: Camera Interface (16-bit Data Bus Mode)

The camera interface is implemented using the GPIO pins of the S1D13513. Several interfaces are implemented in this manner. Therefore, selecting certain combinations of interfaces may not be possible due to the fact that they share the same GPIO pin. For a summary of the possible combinations, refer to Section 5.6, “GPIO Pin Mapping” on page 50.

Before enabling the camera interface, the GPIO pins must be configured appropriately using the GPIO configuration registers (see REG[0C00h] ~ REG[0C0Eh]. For camera interface pin mapping, refer to Section 5.8, “Camera Interface Pin Mapping” on page 51.

To confirm whether the S1D13513 supports a specific camera implementation, see the AC timing details in Section 7.7, “Camera Interface Timing” on page 109.

In the case of 16-bit data bus mode, REG[2010h], Camera Mode Setting Register, should be set to 0001h.

22.1 Programming Flows

The following provides an example programming flow for capturing YUV data from the Camera.

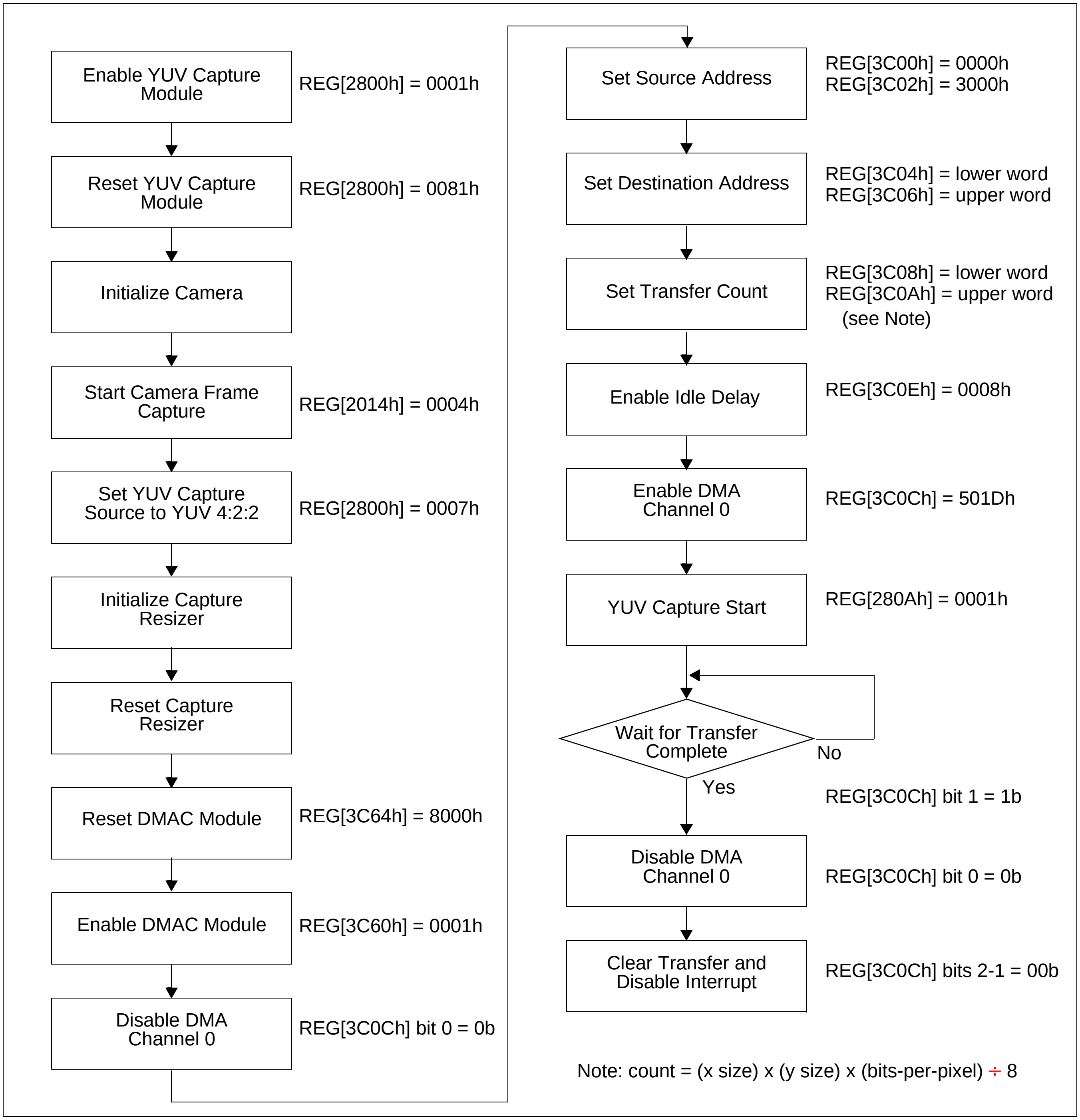


Figure 22-3: YUV Capture from Camera

The following provides an example programming flow for capturing YUV data from Memory.

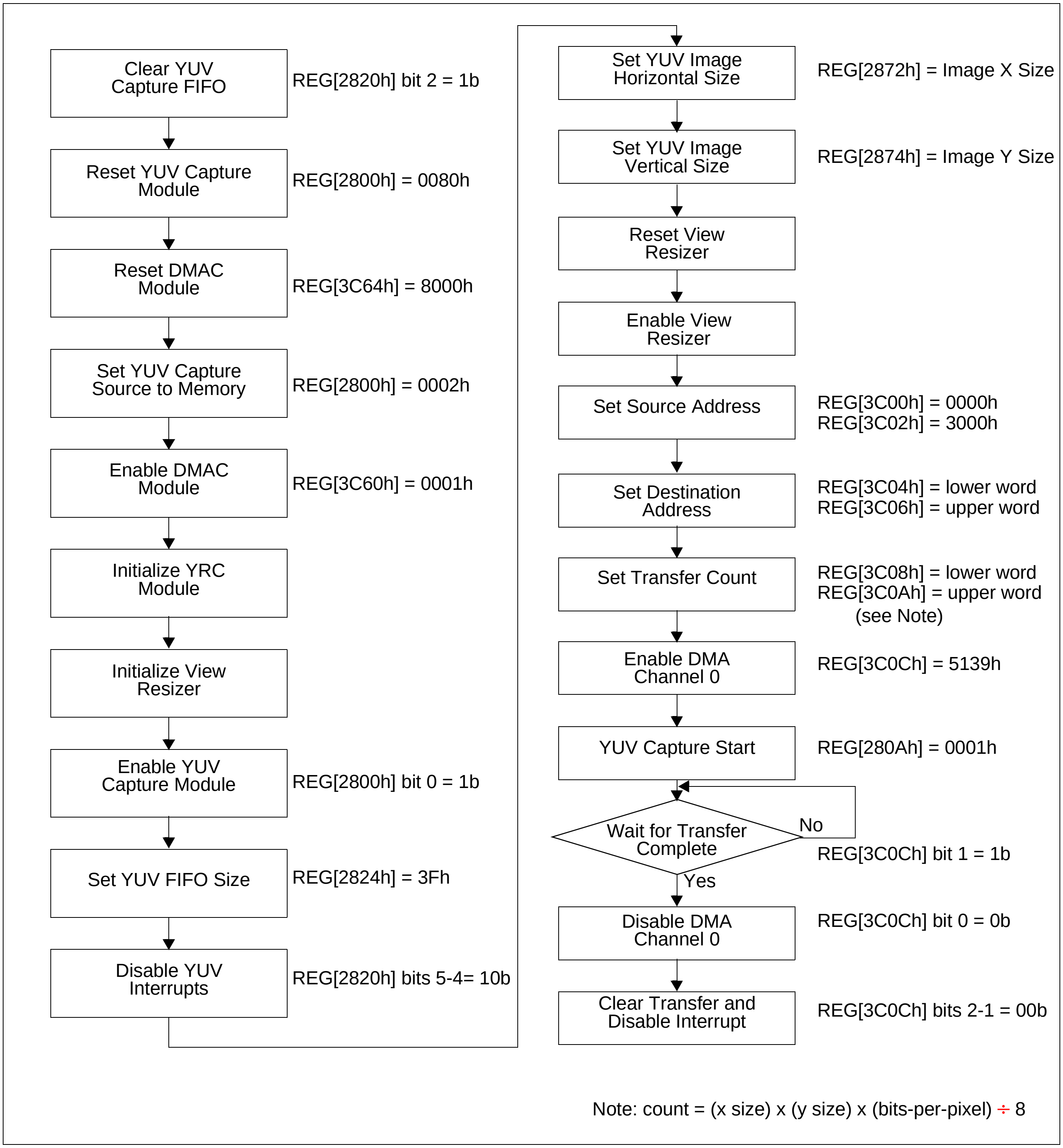
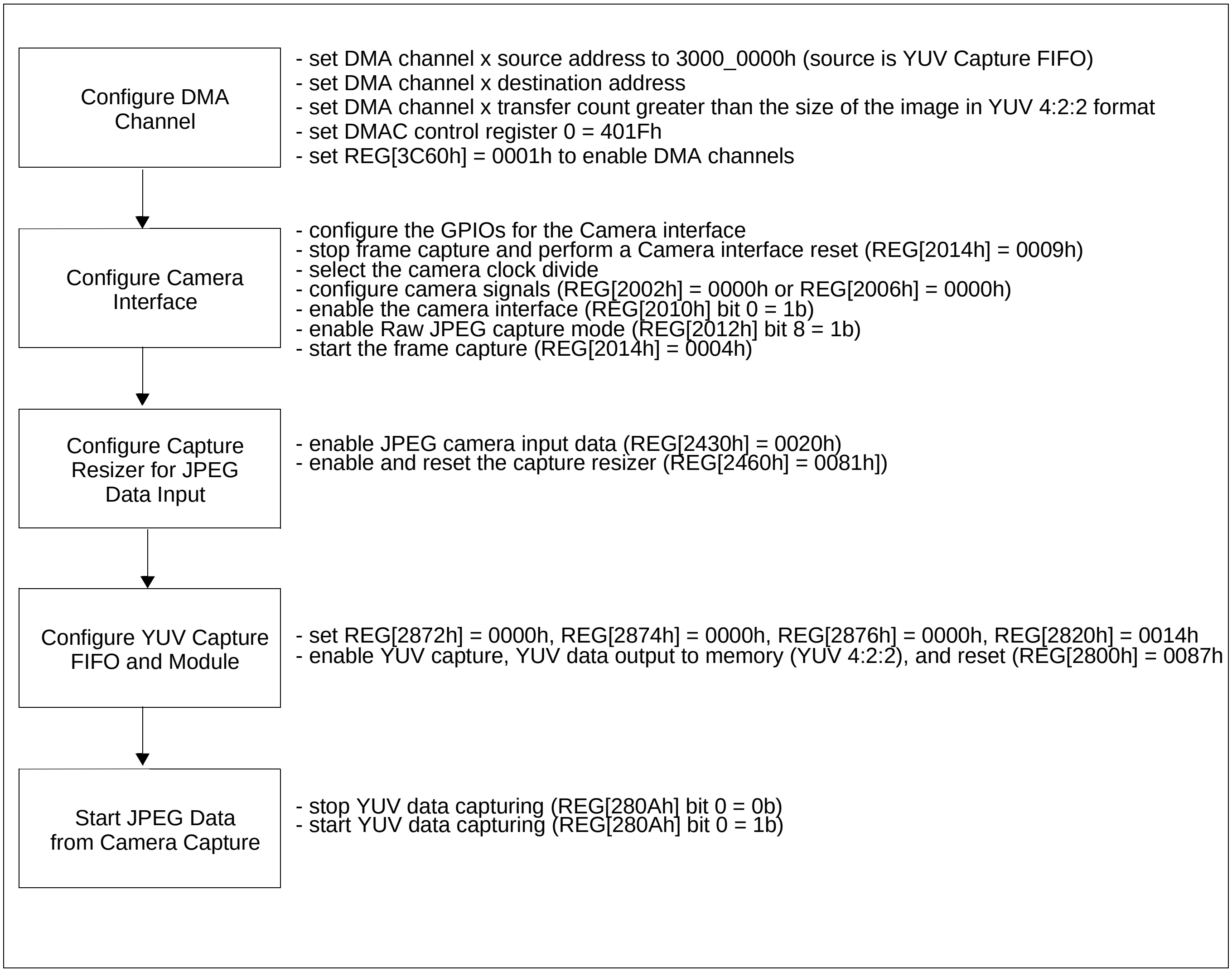


Figure 22-4: YUV Capture from Memory

The following provides an example programming flow for capturing JPEG data from a JPEG capable camera.



Configure DMA Channel

- set DMA channel x source address to 3000_0000h (source is YUV Capture FIFO)
- set DMA channel x destination address
- set DMA channel x transfer count greater than the size of the image in YUV 4:2:2 format
- set DMAC control register 0 = 401Fh
- set REG[3C60h] = 0001h to enable DMA channels

Configure Camera Interface

- configure the GPIOs for the Camera interface
- stop frame capture and perform a Camera interface reset (REG[2014h] = 0009h)
- select the camera clock divide
- configure camera signals (REG[2002h] = 0000h or REG[2006h] = 0000h)
- enable the camera interface (REG[2010h] bit 0 = 1b)
- enable Raw JPEG capture mode (REG[2012h] bit 8 = 1b)
- start the frame capture (REG[2014h] = 0004h)

Configure Capture Resizer for JPEG Data Input

- enable JPEG camera input data (REG[2430h] = 0020h)
- enable and reset the capture resizer (REG[2460h] = 0081h])

Configure YUV Capture FIFO and Module

- set REG[2872h] = 0000h, REG[2874h] = 0000h, REG[2876h] = 0000h, REG[2820h] = 0014h
- enable YUV capture, YUV data output to memory (YUV 4:2:2), and reset (REG[2800h] = 0087h

Start JPEG Data from Camera Capture

- stop YUV data capturing (REG[280Ah] bit 0 = 0b)
- start YUV data capturing (REG[280Ah] bit 0 = 1b)

Figure 22-5: JPEG Capture from Camera

22.2 Frame Capture Interrupt

An interrupt can be generated when capturing of the camera image data is complete. The interrupt generation timing can be synchronized with the strobe output as shown below.

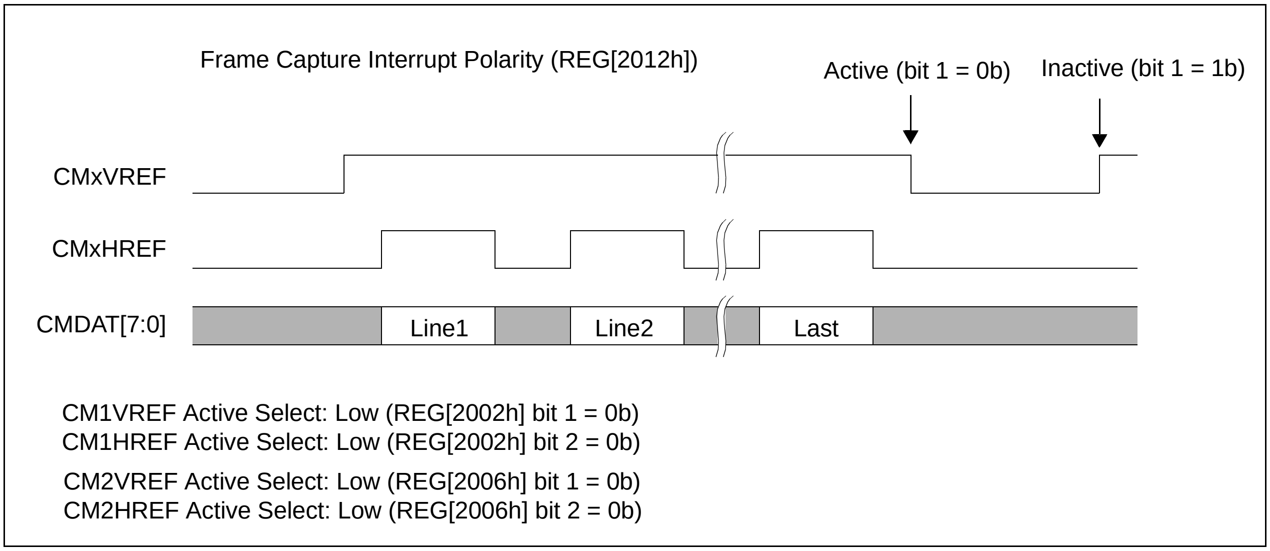

Frame Capture Interrupt Polarity (REG[2012h])
Active (bit 1 = 0b)
Inactive (bit 1 = 1b)
CMxVREF
CMxHREF
CMDAT[7:0]
Line1
Line2
Last
CM1VREF Active Select: Low (REG[2002h] bit 1 = 0b)
CM1HREF Active Select: Low (REG[2002h] bit 2 = 0b)
CM2VREF Active Select: Low (REG[2006h] bit 1 = 0b)
CM2HREF Active Select: Low (REG[2006h] bit 2 = 0b)

Figure 22-6: Frame Capture Interrupt

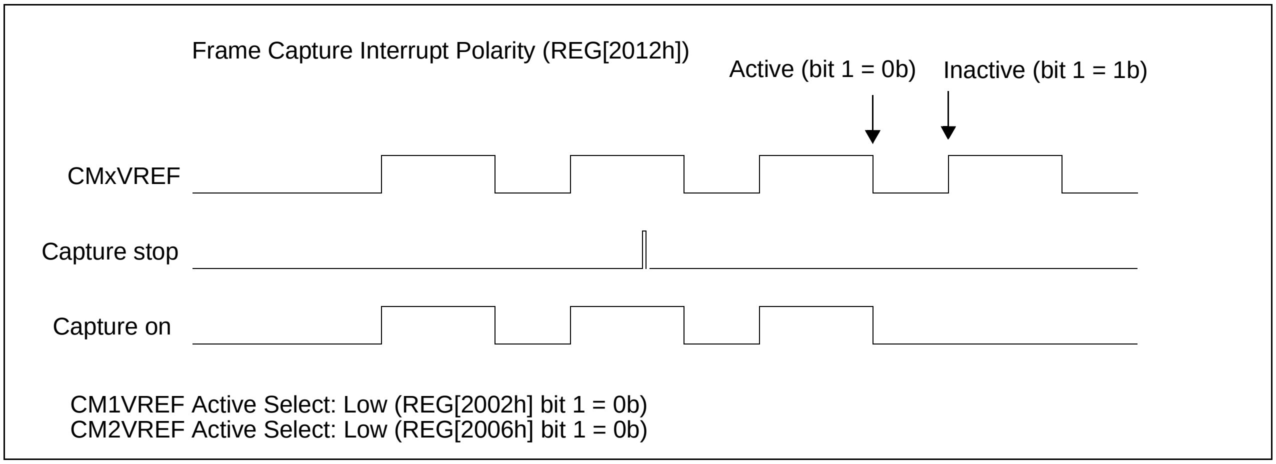

Frame Capture Interrupt Polarity (REG[2012h])
Active (bit 1 = 0b)
Inactive (bit 1 = 1b)
CMxVREF
Capture stop
Capture on
CM1VREF Active Select: Low (REG[2002h] bit 1 = 0b)
CM2VREF Active Select: Low (REG[2006h] bit 1 = 0b)

Figure 22-7: Frame Capture Interrupt (Capture Stops)

22.3 Strobe Control Signal

When the camera interface is enabled, a strobe feature is available. Typically the strobe signal controls the external camera flash or camera data and is used in conjunction with the camera interface to capture or display the optimal camera image after the camera flash has gone off or the camera data output is enabled.

The strobe output is controlled using REG[2020h] ~ REG[2024h]. The strobe control signal output pin is CMSTROUT (see GPIOB7).

22.3.1 Generating a Strobe Pulse

A strobe pulse (CMSTROUT) can be generated in two ways:

Stop Capturing in Repeat Capture Mode

1. Enable the camera interface for continuous frame capture mode (REG[2012h] bit 6 = 0b) and ensure that the CMxVREF and CMxHREF signals are present. ITU-R BT656 data format must not be enabled (REG[2010h] bit 7 = 0b).
2. Configure the Strobe Line Delay Timing (REG[2020h]), Strobe Pulse Width (REG[2022h], Strobe Active Select (REG[2024h] bit 1), and Strobe Capture Delay (REG[2024h] bits 7-4).
3. Enable the strobe signal (CMSTROUT) by stopping the camera frame capture (REG[2014h] bit 3 = 1b). The last camera frame captured depends on the Strobe Capture Delay Control in step 2.

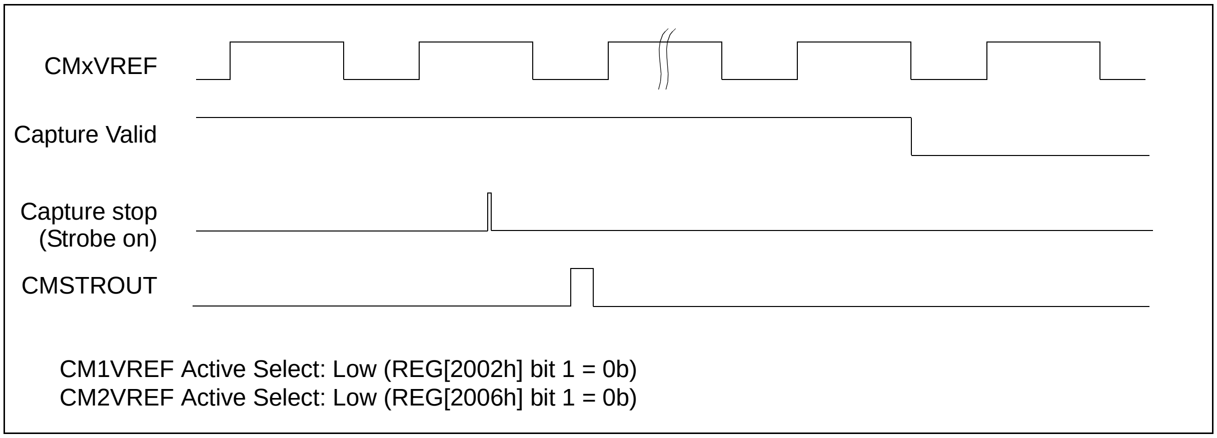

CMxVREF
Capture Valid
Capture stop
(Strobe on)
CMSTROUT
CM1VREF Active Select: Low (REG[2002h] bit 1 = 0b)
CM2VREF Active Select: Low (REG[2006h] bit 1 = 0b)

Figure 22-8: Strobe Operation (Continuous Capture Stopped)

Single Camera Frame Capture

1. Enable the camera interface for single frame capture mode (REG[2012h] bit 6 = 1b) and ensure that the CMxVREF and CMxHREF signals are present. ITU-R BT656 data format must not be enabled (REG[2010h] bit 7 = 0b).
2. Configure the Strobe Line Delay Timing (REG[2020h]), Strobe Pulse Width (REG[2022h], Strobe Active Select (REG[2024h] bit 1), and Strobe Capture Delay (REG[2024h] bits 7-4).
3. Enable the strobe signal (CMSTROUT) by capturing a camera frame (REG[2014h] bit 2 = 1b). The camera frame that is captured, is the one occurring right after the strobe signal and is not dependant on the Strobe Capture Delay in step 2.

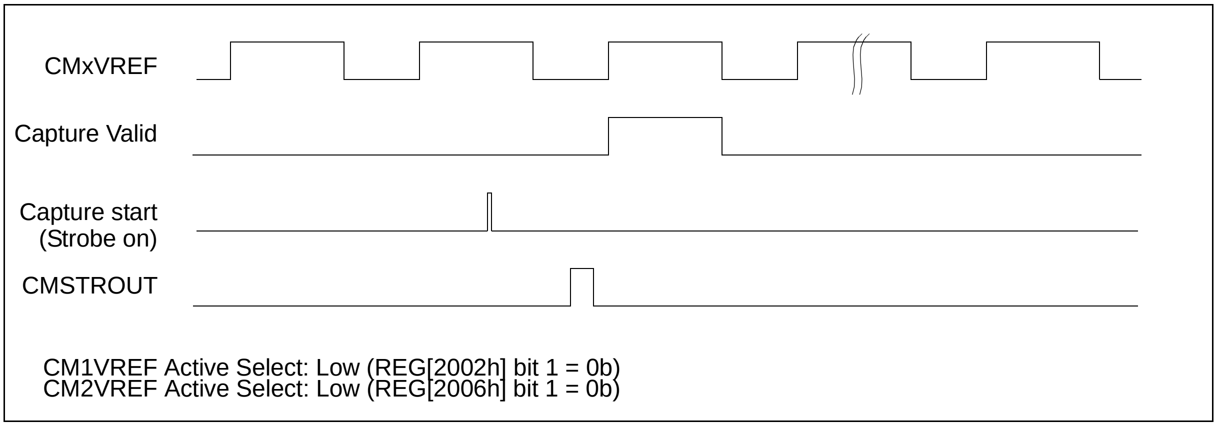

CMxVREF
Capture Valid
Capture start
(Strobe on)
CMSTROUT
CM1VREF Active Select: Low (REG[2002h] bit 1 = 0b)
CM2VREF Active Select: Low (REG[2006h] bit 1 = 0b)

Figure 22-9: Strobe Operation (Single Frame Capture)

22.3.2 Strobe Timing

The strobe pulse (CMSTROUT) begins on the falling edge of CMxHREF after CMxVREF as specified by the Strobe Line Delay Timing bits (REG[2020h] bits 15-0). A zero delay (REG[2020h] bits 15-0 = 00h) starts the strobe pulse (CMSTROUT) on the first falling edge of CMxHREF after CMxVREF.

Note

Both the Line Delay and Pulse Width signals are specified by counting HREFs which leads to an inherent timing delay if the HREF signal stops. This inherent delay must be considered when programming the Line Delay (REG[2020h]) and Pulse Width (REG[2022h]) registers.

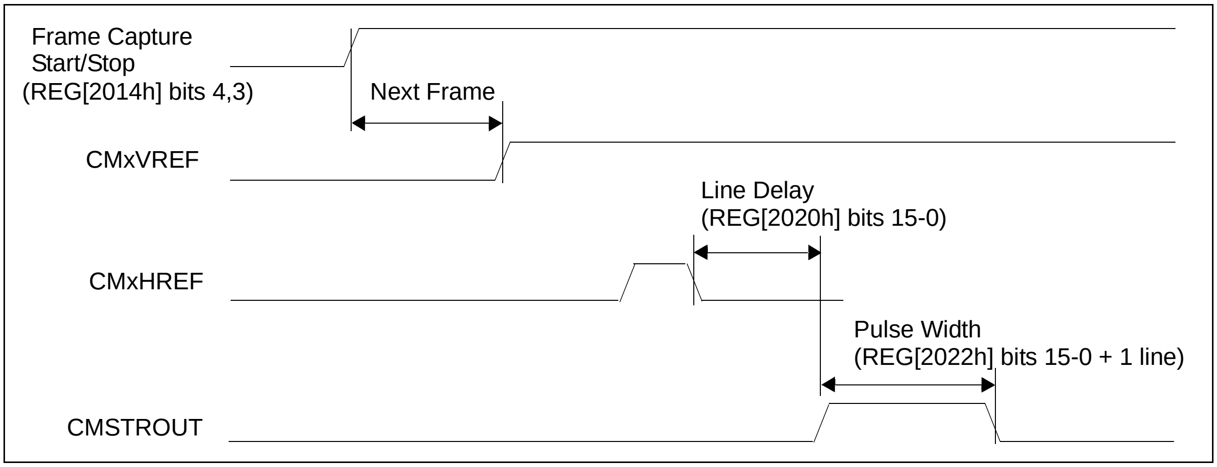


Figure 22-10: Strobe Signal Output Timing

Note

The Line Delay (REG[2020h] bits 15-0) and the Pulse Width (REG[2022h] bits 15-0) may be set greater than the period of the CMxVREF signal.

23 I2C Interface

The S1D13513 includes an I2C interface which is typically used to program the camera. The interface supports master mode only and can be configured at two speeds: standard (100kbs) and fast (400kbs).

The I2C interface is implemented using the GPIO pins of the S1D13513. Several interfaces are implemented in this manner, however, the S1D13513 has been designed so that the I2C interface pins do not share function with any other interfaces. For a summary of the GPIO pin combinations, refer to Section 5.6, “GPIO Pin Mapping” on page 50.

Before enabling the I2C interface, the GPIO pins must be configured appropriately using the GPIO configuration registers (see REG[0C06h] bits 13-10). For I2C interface pin mapping, refer to Section 5.9, “I2C Interface Pin Mapping” on page 52.

To confirm whether the S1D13513 supports a specific camera implementation, see the AC timing details in Section 7.9, “I2C Interface Timing” on page 117.

23.1 Programming Flows

The following provides an example programming flow to write N byte(s) to an I2C device (8-bit data device).

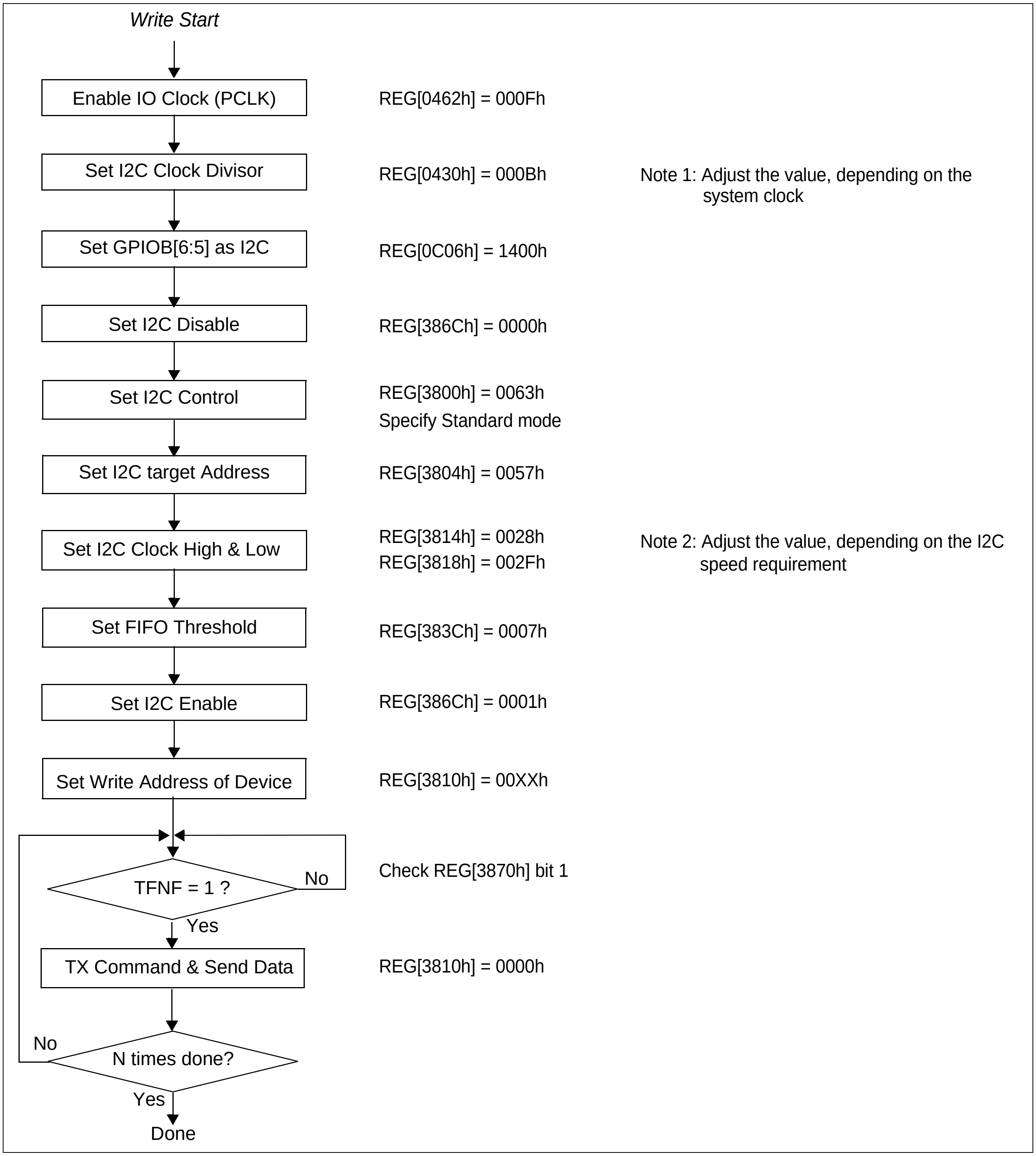


Figure 23-1: I2C Interface Write Flow Example

The following provides an example programming flow to read N byte(s) from an I2C device (8-bit data device).

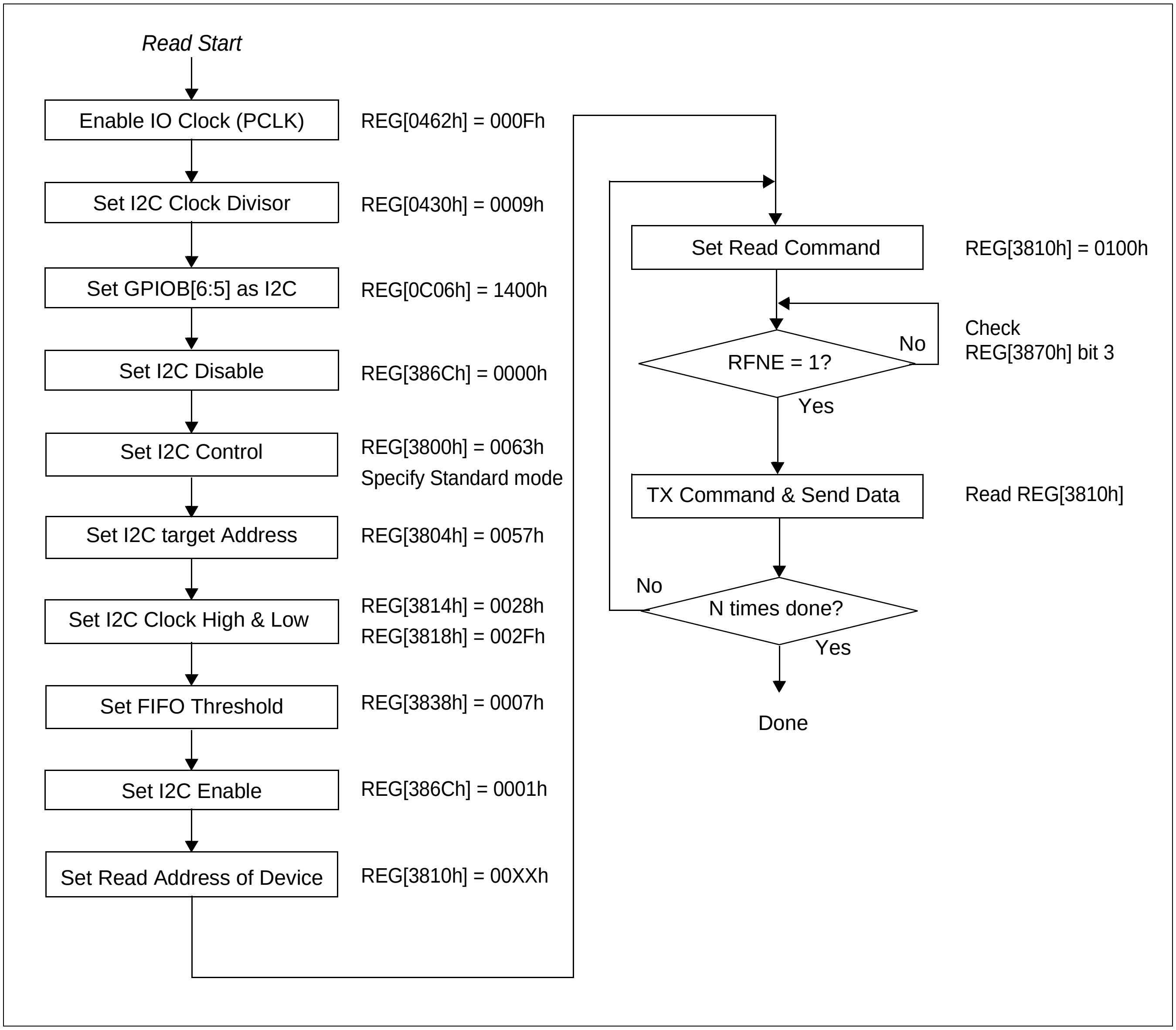


Figure 23-2: I2C Interface Read Flow Example

The following provides an example programming flow for disabling the I2C Module.

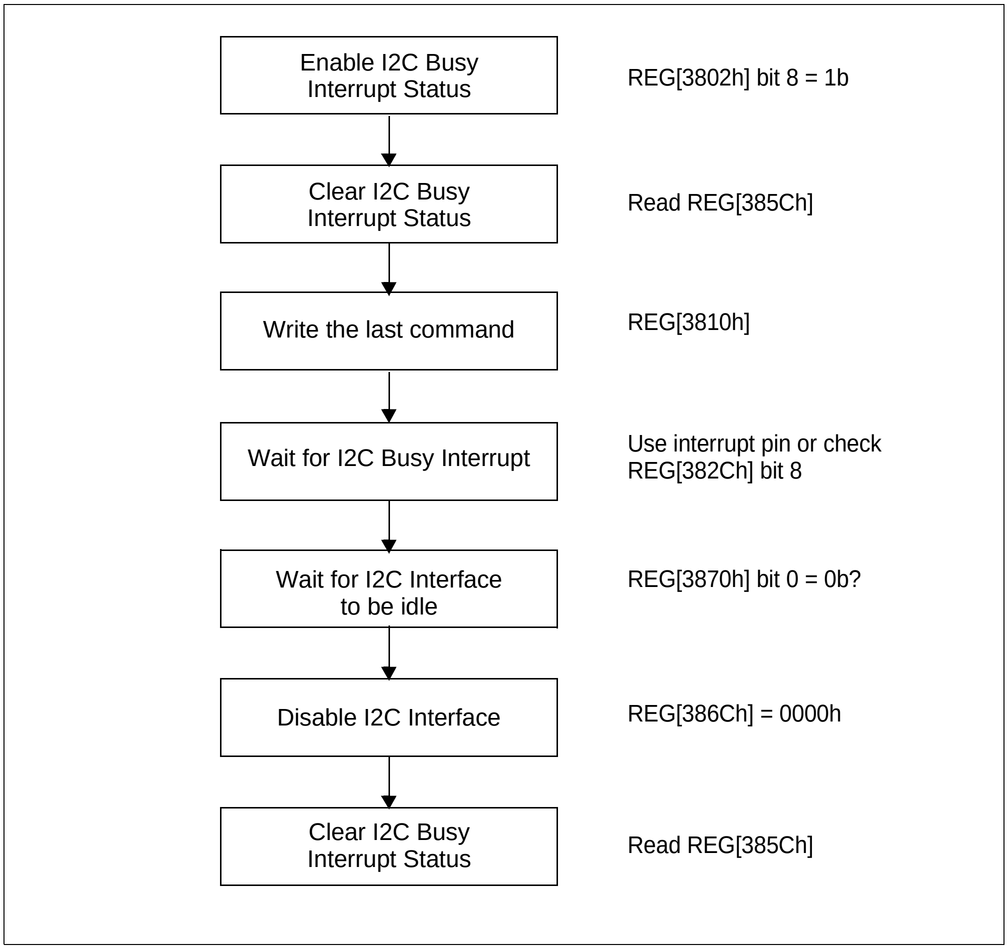

Enable I2C Busy Interrupt Status
REG[3802h] bit 8 = 1b
Clear I2C Busy Interrupt Status
Read REG[385Ch]
Write the last command
REG[3810h]
Wait for I2C Busy Interrupt
Use interrupt pin or check REG[382Ch] bit 8
Wait for I2C Interface to be idle
REG[3870h] bit 0 = 0b?
Disable I2C Interface
REG[386Ch] = 0000h
Clear I2C Busy Interrupt Status
Read REG[385Ch]

Figure 23-3: I2C Module Disable Sequence

24 Keypad Interface

The S1D13513 is designed with a keypad interface supporting up to a 5x5 matrix.

Note
The keypad interface is not available for the QFP package.

24.1 Keypad Interface Example Connection

The keypad interface uses two sets of pins to signal data to the S1D13513. KEYX[4:0] and KEYY[4:0] are used to determine which key of the 5x5 matrix is pressed. Both sets of pins are multiplexed on GPIO pins with KEYX[4:0] on GPIOA[4:0] and KEYY[4:0] on GPIOB[4:0]. The GPIO pins must be configured for use by the keypad interface before the keypad interface is enabled. For a summary of GPIO pin usage, see Section 5.6, “GPIO Pin Mapping” on page 50.

To avoid unintended keypad input due to reverse current, diodes must be placed between the KEYY[4:0] pins and the keypad as shown below.

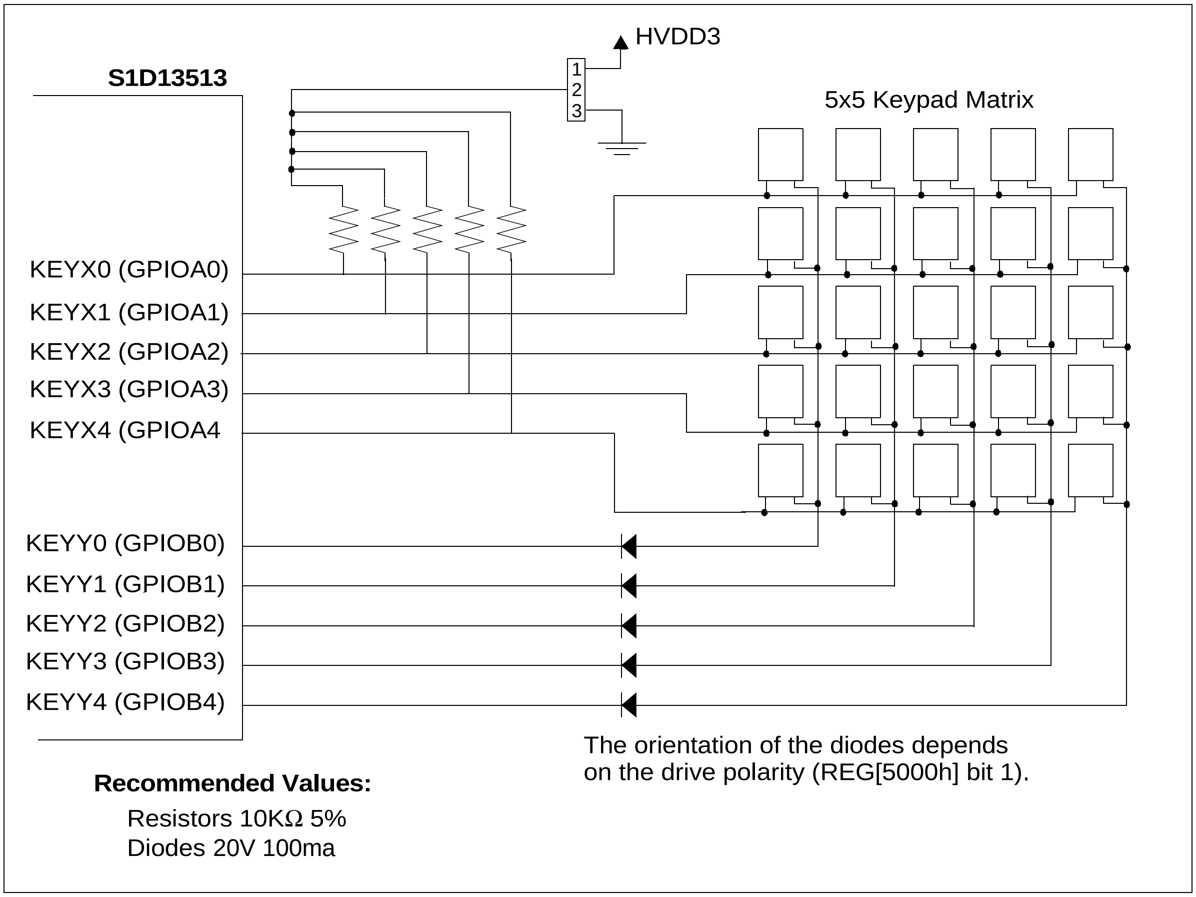


Figure 24-1: Keypad Interface Example Connection

24.2 Key Scanning

The S1D13513 scans for key presses from 1-25 as shown in the figure below. The interface is designed to detect one keypress at a time, however, keys within the same column (e.g. 1, 6, 11, 16, 21) can be detected simultaneously.

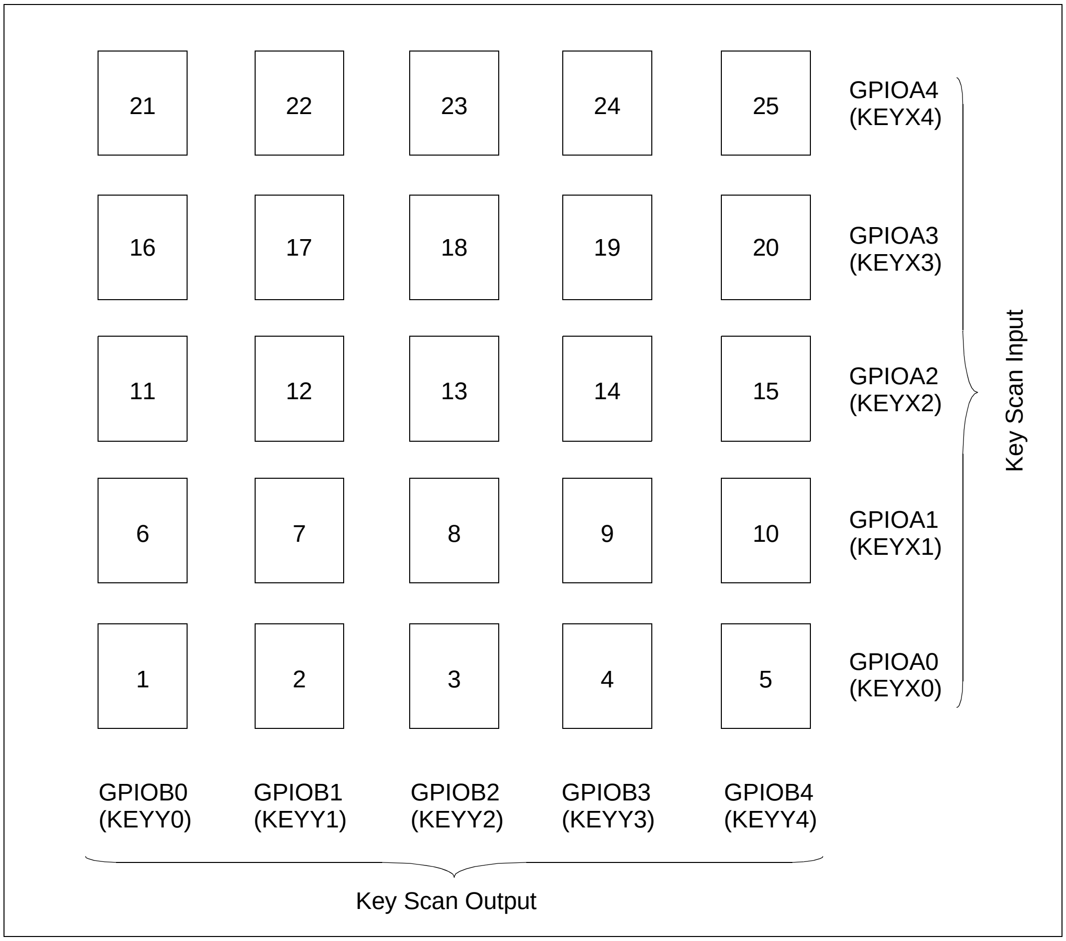


Figure 24-2: Keypad Interface Example Connection

Note

Once a key press is detected, key scanning must be re-started by software.

The key scan proceeds as an active low scan of GPIOB0 through GPIOB4. Key input is recognized when a Low input occurs on a GPIOA pin. The following table shows the applicable values for the GPIO pins and Key Scan Data register (REG[5004h]) for each specific key press.

Table 24-1 : Key Press Values

Key Press	GPIOB[4:0]	GPIOA[4:0]	REG[5004h]	REG[5004h]
“1”	1Eh	1Eh	1E01h	0101h
“2”	1Dh	1Eh	1D01h	0201h
“3”	1Bh	1Eh	1B01h	0401h
“4”	17h	1Eh	1701h	0801h
“5”	0Fh	1Eh	0F01h	1001h
“6”	1Eh	1Dh	1E02h	0102h
“7”	1Dh	1Dh	1D02h	0202h
“8”	1Bh	1Dh	1B02h	0402h
“9”	17h	1Dh	1702h	0802h
“10”	0Fh	1Dh	0F02h	1002h
“11”	1Eh	1Bh	1E04h	0104h
“12”	1Dh	1Bh	1D04h	0204h
“13”	1Bh	1Bh	1B04h	0404h
“14”	17h	1Bh	1704h	0804h
“15”	0Fh	1Bh	0F04h	1004h
“16”	1Eh	17h	1E08h	0108h
“17”	1Dh	17h	1D08h	0208h
“18”	1Bh	17h	1B08h	0408h
“19”	17h	17h	1708h	0808h
“20”	0Fh	17h	0F08h	1008h
“21”	1Eh	0Fh	1E10h	0110h
“22”	1Dh	0Fh	1D10h	0210h
“23”	1Bh	0Fh	1B10h	0410h
“24”	17h	0Fh	1710h	0810h
“25”	0Fh	0Fh	0F10h	1010h

24.3 Example Programming Flows

The following programming flow provides an example sequence for initializing the Keypad interface.

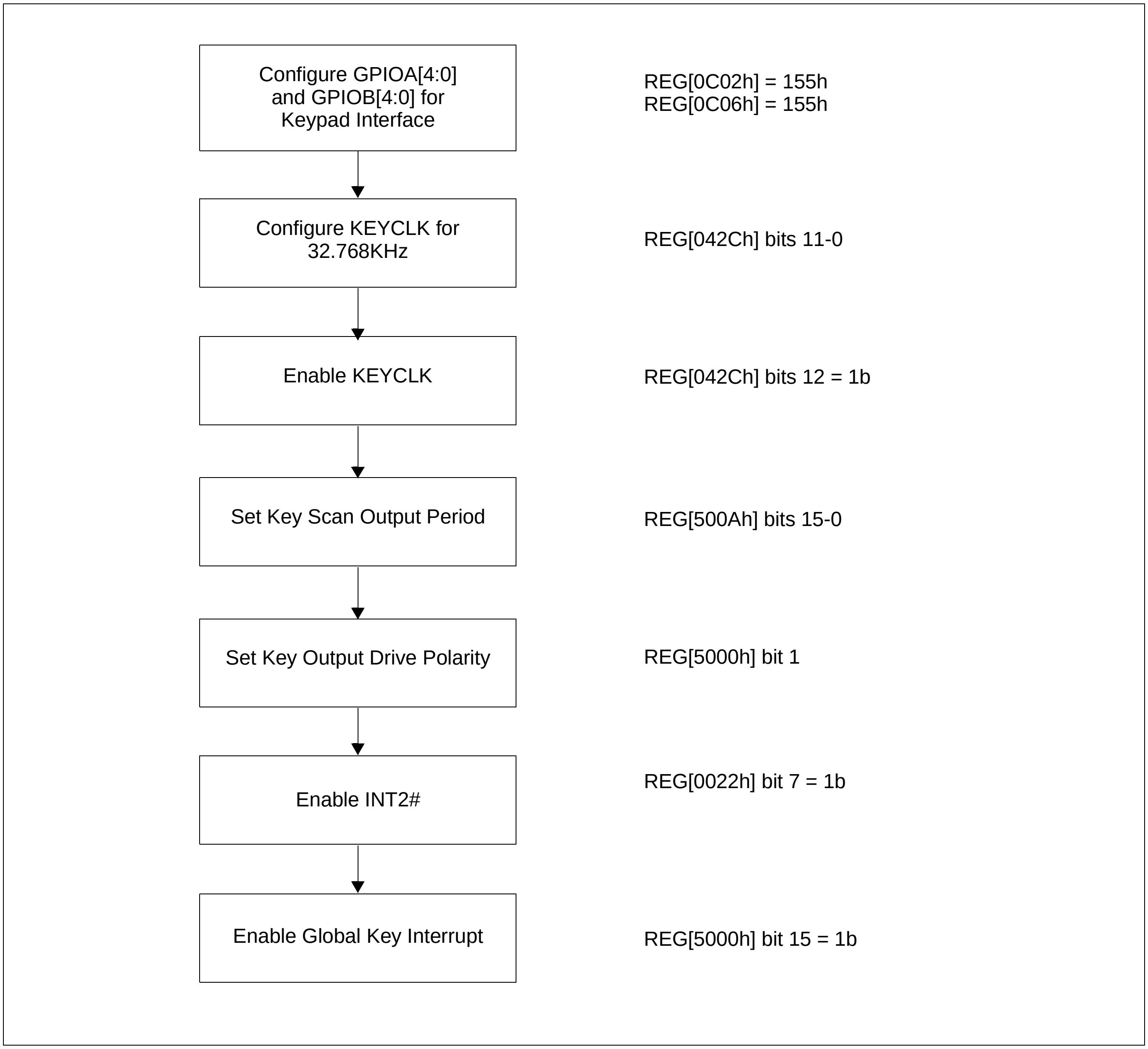


Figure 24-3: Keypad Interface Initialization Programming Flow

The following programming flow shows an example method for detecting key presses using the Keypad interface.

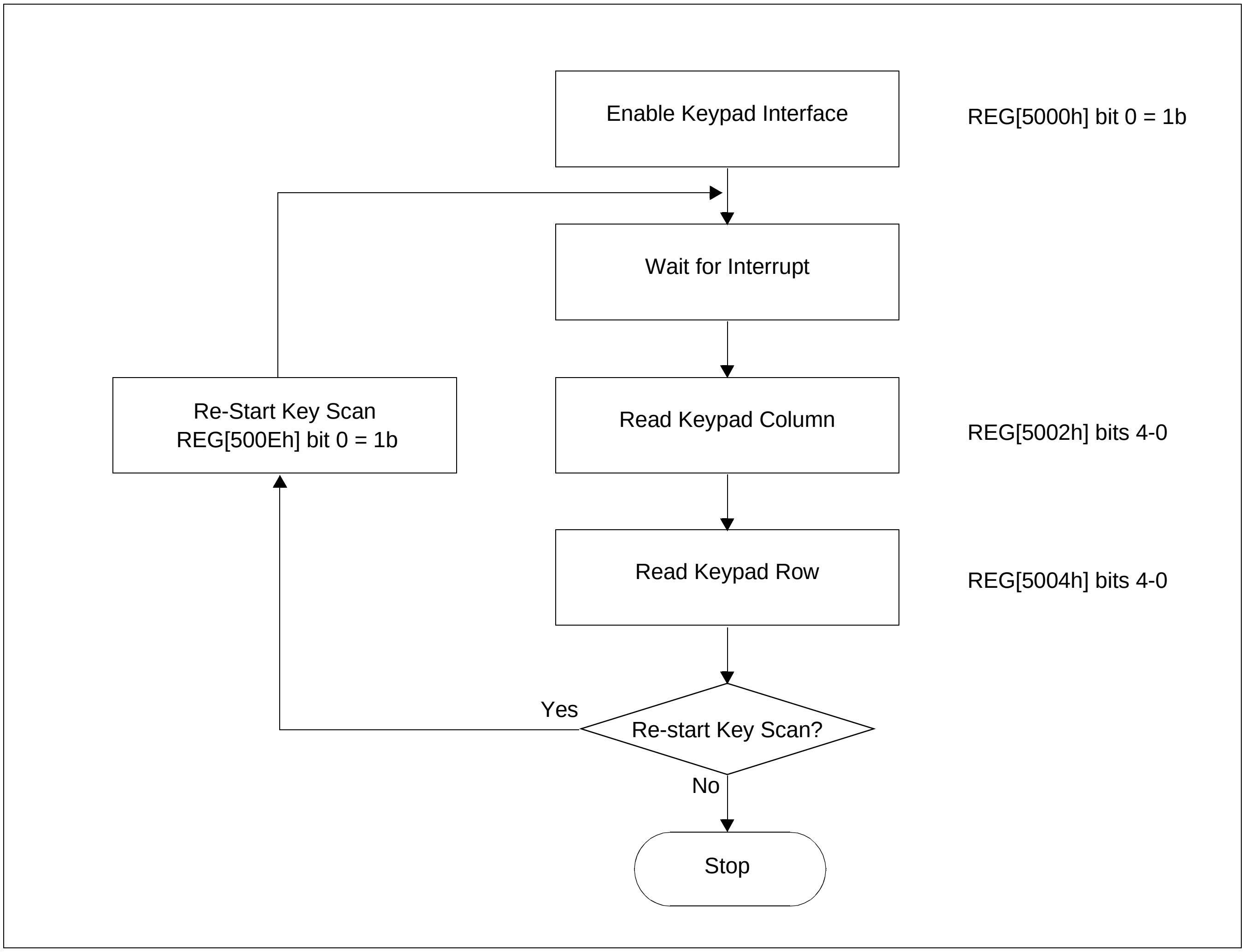


Figure 24-4: Keypad Interface Key Detect Programming Flow

25 Crystal Oscillator Circuit

The S1D13513 is designed with two crystal oscillation circuits which can be used as clock sources. For details on the clocks, see Section 9, “Clocks” on page 124.

The oscillation characteristics depend on the components used in the circuit (i.e. X’tal, Rf, Rd, Cg, Cd, and board characteristics). An example circuit and example component values are shown below. Actual values used must be confirmed for each implementation.

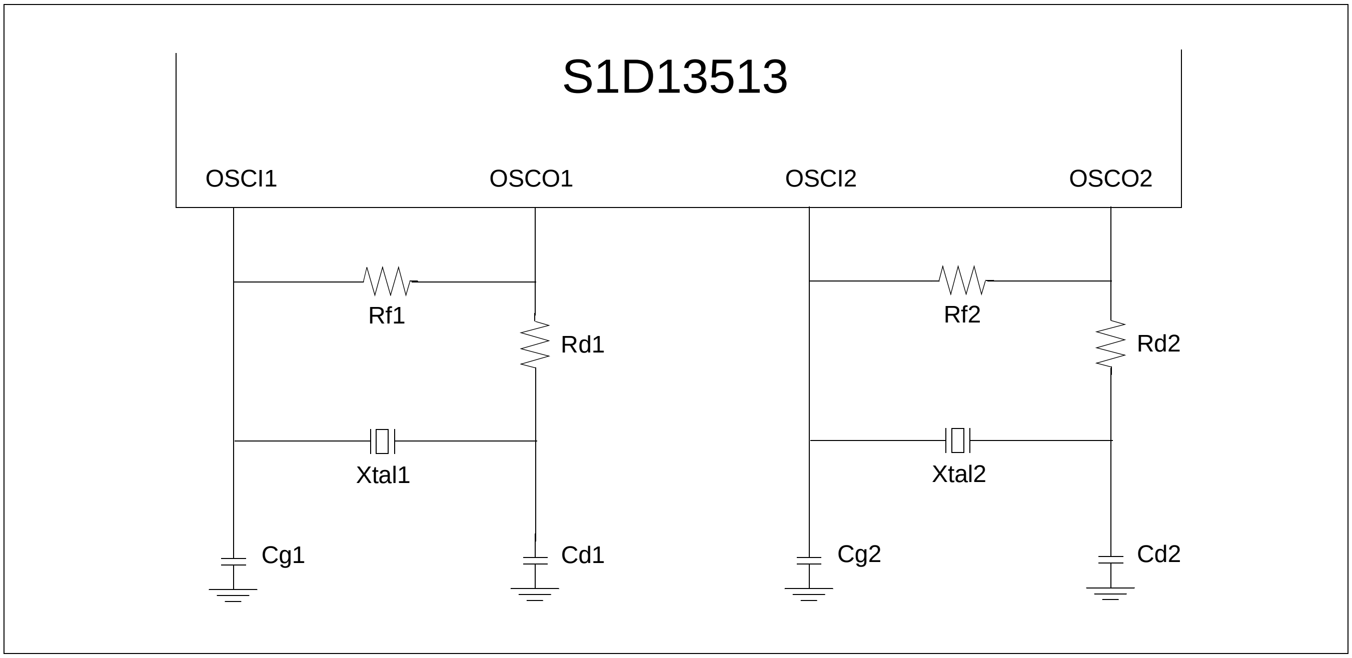


Figure 25-1: Recommended Crystal Oscillator External Circuit

Table 25-1 : Recommended Crystal Oscillator External Circuit Parameters

Symbol	Parameter	Min	Typ	Max	Units
Rf1	Rf1	—	1	—	MΩ
Rd1	Rd1	—	0	—	Ω
Cg1	Cg1	—	10	—	pF
Cd1	Cd1	—	10	—	pF
Xtal1	Fundamental mode crystal	5	—	20	MHz
Rf2	Rf2	—	1	—	MΩ
Rd2	Rd2	—	0	—	Ω
Cg2	Cg2	—	5	—	pF
Cd2	Cd2	—	5	—	pF
Xtal2	Fundamental mode crystal	5	—	27	MHz

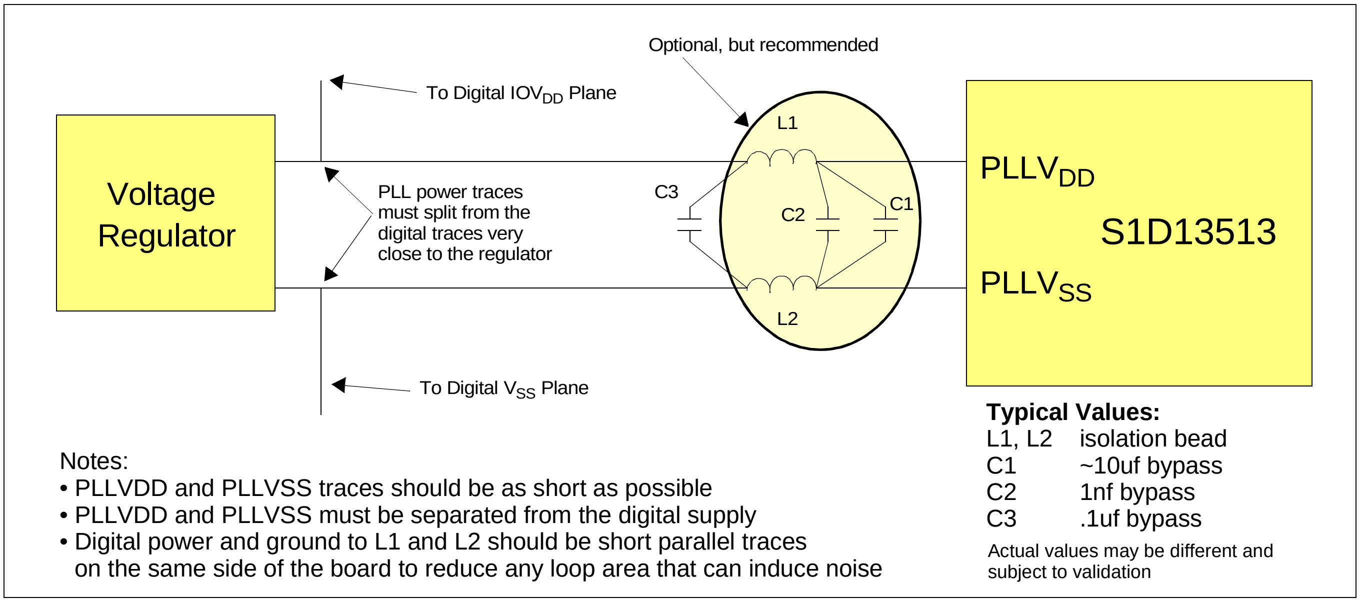
26 Design Considerations

26.1 Guidelines for PLL Power Layout

The PLL circuit is an analog circuit and is very sensitive to noise on the input clock waveform or the power supply. Noise on the clock or the supplied power may cause the operation of the PLL circuit to become unstable or increase the jitter.

Due to these noise constraints, it is highly recommended that the power supply traces or the power plane for the PLL be isolated from those of other power supplies. Filtering should also be used to keep the power as clean as possible.

The following are guidelines which, if followed, will result in cleaner power to the PLL, resulting in a cleaner and more stable clock. Even a partial implementation of these guidelines will give results.


Optional, but recommended
To Digital IOV$_{DD}$ Plane
Voltage Regulator
PLL power traces must split from the digital traces very close to the regulator
C3
L1
C2
C1
L2
PLLV$_{DD}$
S1D13513
PLLV$_{SS}$
To Digital V$_{SS}$ Plane

Typical Values:

L1, L2	isolation bead
C1	~10uf bypass
C2	1nf bypass
C3	.1uf bypass

Actual values may be different and subject to validation

Notes:

- PLLVDD and PLLVSS traces should be as short as possible
- PLLVDD and PLLVSS must be separated from the digital supply
- Digital power and ground to L1 and L2 should be short parallel traces on the same side of the board to reduce any loop area that can induce noise

Figure 26-1: PLL Power Layout

- Place the ferrite beads (L1 and L2) parallel to each other with minimal clearance between them. Both bypass caps (C2 and C3) should be as close as possible to the inductors. The traces from C3 to the power planes should be short parallel traces on the same side of the board with just the normal small clearance between them. Any significant loop area here will induce noise. If there is a voltage regulator on the board, try to run these power traces directly to the regulator instead of dropping to the power planes (still follow above rules about parallel traces).

- The analog ground point where bypass cap (C2) connects to the ground isolation inductor (L1) becomes the analog ground central point for a ground star topology. None of the components connect directly to the analog ground pin of the S1D13513 ($PLLV_{SS}$) except for a single short trace from C2 to the $PLLV_{SS}$ pin. The ground side of the large bypass capacitor (C1) should also have a direct connection to the star point.
- The same star topology rules used for analog ground apply to the analog power connection where L2 connects to C2.
- All of the trace lengths should be as short as possible.
- If possible, have all the PLL traces on the same outside layer of the board. The only exception is C1, which can be put on the other side of the board if necessary. C1 does not have to be as close to the analog ground and power star points as the other components.
- If possible, include a partial plane under the PLL area only (area under PLL components and traces). The solid analog plane should be grounded to the C2 (bypass) pad. This plane won’t help if it is too large. It is strictly an electrostatic shield against coupling from other layers’ signals in the same board area. If such an analog plane is not possible, try to have the layer below the PLL components be a digital power plane instead of a signal layer.
- If possible, keep other board signals from running right next to PLL pin vias on any layer.
- Wherever possible use thick traces, especially with the analog ground and power star connections to either side of C2. Try to make them as wide as the component pads – thin traces are more inductive.

It is likely that manufacturing rules will prohibit routing the ground and power star connections as suggested. For instance, four wide traces converging on a single pad could have reflow problems during assembly because of the thermal effect of all the copper traces around the capacitor pad. One solution might be to have only a single trace connecting to the pad and then have all the other traces connecting to this wide trace a minimum distance away from the pad. Another solution might be to have the traces connect to the pad, but with thermal relief around the pad to break up the copper connection. Ultimately the board must also be manufacturable, so best effort is acceptable.

27 Mechanical Data

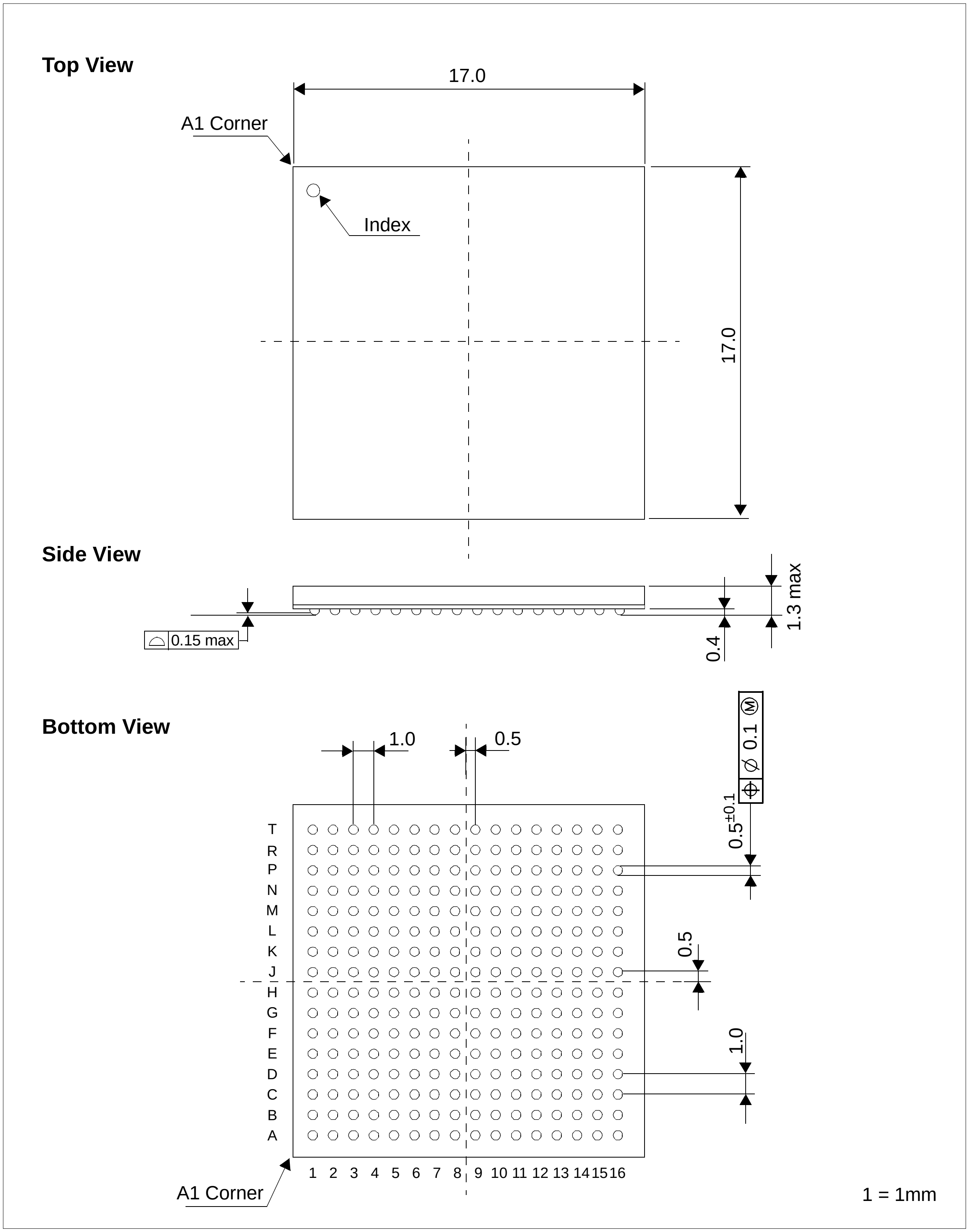
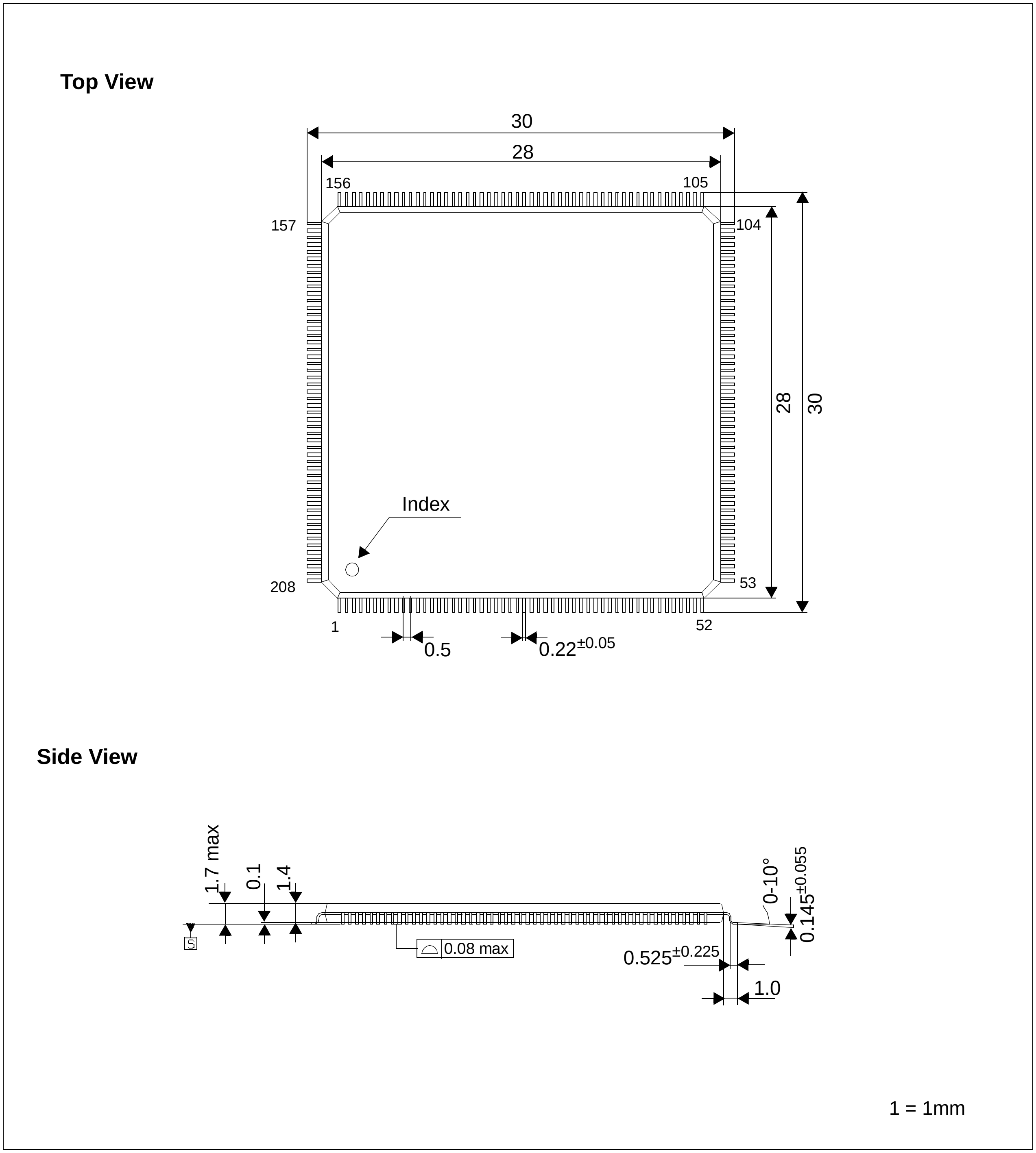


Figure 27-1: S1D13513 PBGA 256-pin Package



Top View

Side View

Figure 27-2: S1D13513 QFP22 208-pin Package

28 References

The following documents contain additional information related to the S1D13513. Document numbers are listed in parenthesis after the document name. All documents can be found at the Epson Research and Development Website at **www.erd.epson.com**.

- S1D13513 Product Brief (X78B-C-001-xx)
- S5U13513P00C100 Evaluation Board User Manual (X78A-G-003-xx)

29 Sales and Technical Support

AMERICA

EPSON ELECTRONICS AMERICA, INC.
2580 Orchard Parkway
San Jose, CA 95131,USA
Phone: +1-800-228-3964 FAX: +1-408-922-0238

EUROPE

EPSON EUROPE ELECTRONICS GmbH
Riesstrasse 15, 80992 Munich,
GERMANY
Phone: +49-89-14005-0 FAX: +49-89-14005-110

ASIA

EPSON (CHINA) CO., LTD.
7F, Jinbao Bldg., No.89 Jinbao St.,
Dongcheng District,
Beijing 100005, CHINA
Phone: +86-10-8522-1199 FAX: +86-10-8522-1125

SHANGHAI BRANCH
7F, Block B, High-Tech Bldg., 900, Yishan Road,
Shanghai 200233, CHINA
Phone: +86-21-5423-5577 FAX: +86-21-5423-4677

SHENZHEN BRANCH
12F, Dawning Mansion, Keji South 12th Road,
Hi-Tech Park, Shenzhen 518057, CHINA
Phone: +86-755-2699-3828 FAX: +86-755-2699-3838

EPSON HONG KONG LTD.
20/F, Harbour Centre, 25 Harbour Road
Wanchai, Hong Kong
Phone: +852-2585-4600 FAX: +852-2827-4346
Telex: 65542 EPSCO HX

EPSON TAIWAN TECHNOLOGY & TRADING LTD.
14F, No. 7, Song Ren Road,
Taipei 110, TAIWAN
Phone: +886-2-8786-6688 FAX: +886-2-8786-6660

EPSON SINGAPORE PTE., LTD.
1 HarbourFront Place,
#03-02 HarbourFront Tower One, Singapore 098633
Phone: +65-6586-5500 FAX: +65-6271-3182

SEIKO EPSON CORP.
KOREA OFFICE
50F, KLI 63 Bldg., 60 Yoido-dong
Youngdeungpo-Ku, Seoul, 150-763, KOREA
Phone: +82-2-784-6027 FAX: +82-2-767-3677

SEIKO EPSON CORP.
SEMICONDUCTOR OPERATIONS DIVISION

IC Sales Dept.
IC International Sales Group
421-8, Hino, Hino-shi, Tokyo 191-8501, JAPAN
Phone: +81-42-587-5814 FAX: +81-42-587-5117

29.1 Ordering Information

To order the S1D13513 Display Controller, contact the Epson sales representative in your area.

Change Record

X78C-A-001-01 Revision 1.7 - Issued: July 28, 2010

- section 10.4.7 Sprite Registers - update first sentence in second and thiird paragraphs to clearify description
- section 10.4.7 Sprite Registers - under Sprite Registers when Sprite Engine is Enabled, update first sentence to clearify description
- section 10.4.7 Sprite Registers - in figure 10-2, *Sprite #0 - #15 Register Mapping*, move Register references down and change REG[1600h] to REG[15FFh] to clearify description

X78C-A-001-01 Revision 1.6 - Issued: May 20, 2010

- section 9.1 Clock Overview - remove WDTCLK and associated circuitry from figure 9-1 Clock Diagram

X78C-A-001-01 Revision 1.5 - Issued: August 20, 2009

- all changes from the last revision of the spec are highlighted in Red
- section 5.8 Camera Interface Pin Mapping - add table 5-21 *Camera Interface Pin Mapping (16-bit data bus mode - PBGA 256-pin Package)*
- section 7.1.1 Input Clocks - change tf and tr to” 0.2 Ts” in figure 7-1, *Clock Requirements for OSC1/OSC2/CLKI3/BUSCLK when used as Clock Input*, and Figure 7-2, *Clock Requirements for OSC1/OSC2 when used as Crystal Oscillator Input*
- section 7.6.1 Generic TFT Panel Timing - in Generic RGB Type Interface Panel Horizontal Timing, multiple changes to Table 7-34, *Generic RGB Type Interface Panel Horizontal Timing*
- section 7.7.1 Camera Interface YUV Timing (8-bit Data Bus Mode) - add 8-bit Data Bus Mode to section title and figures and tables
- section 7.7.2 Camera Interface YUV Timing (16-bit Data Bus Mode) - add this section
- REG[001Ah] - add this register
- REG[1808h] bits 2-0 - add note “When Move BitBLT with color expansion is selected...”
- REG[1826h] - changes to note 2 and add note 3 “When Move BitBLT with Color Expand is selected...”
- REG[2002h] bits 4-3 - add YUV Data Format (16-bit format) column to table in bit description
- REG[2002h] bit 2 - correct typo in bit description, change “Camera” to “Camera1”
- section 22 Camera Interface - add 16-bit Data Bus Mode to section

X78C-A-001-01 Revision 1.4 - Issued: February 16, 2009

- all changes from the last revision of the spec are highlighted in Red
- REG[1810h] ~ REG[181Eh], add note “For 8 bpp operations, the BitBLT must begin on an even boundary...”
- REG[2000h], clarified that table is for “Camera1”
- REG[3C08h] ~ REG[3C0Ah], updated typo in bit description to change “224” to “2^24”
- REG[3C18h] ~ REG[3C1Ah], updated typo in bit description to change “224” to “2^24”
- section 22.1, clarified formula in YUV Capture from Camera and YUV Capture from Memory figures

X78C-A-001-01 Revision 1.3 - Issued: February 09, 2009

- all changes from the last revision of the spec are highlighted in Red
- created from X78B-A-001-xx Revision 1.2
- Globally remove Freescale MPC556
- Globally remove Freescale MPC821
- section 2.4, for the display features section removed ARGB 8:5:6:5 from the alpha blending options
- section 5.2.1 Host Interface - rewrite BURST# pin description in table 5-3
- section 5.2.1 Host Interface - rewrite BDIP# pin description in table 5-3
- section 5.4 Host Bus Interface Pin Mapping - add note 4 to table
- section 7.1.3, updated PLL max output frequency to 130MHz
- section 7.4.8, added a note regarding SH3 active low WAIT# always driven mode for S1D13513 revisions 00h and 01h
- section 7.4.12, for the MPC555 Read Timing table changed t19min parameter from “0ns” to “-20ns”
- section 7.4.16 ~ 7.4.17, changed references from “CS#” to “SCS#” in the Serial Host Timing
- section 7.4.16 ~ 7.4.17, for the Serial Host Timing changed the t2min parameter (SCK Period) from “30ns” to “63ns”
- REG[0000h], changed default register value for the Product ID Register 0 from “0100h” to “0200h”
- REG[0020h] bit 6, updated the register reference to clear this bit from “REG[2016h] bit 1” to “REG[20014h] bit 1”
- REG[0044h] bit 15, un-reserved this register and added the Prefetch Buffer Disable bit and bit description

- REG[040Ch], updated the note to “...value of 8311h results in a PLL1 output of 100MHz when a 50MHz reference clock is input to PLL1.”
- REG[0414h], clarified the wording of the note for the example PLL configuration value
- REG[0836h] bit 13, removed the “÷2” from the end of the formula
- REG[0836h] bits 10-9, removed the “÷2” from the end of the formula
- REG[102Ch], fixed typo in bit description formula
- REG[102Eh], fixed typo in bit description formula
- REG[1862h], updated the description for the 1-bit source bit format
- REG[244Eh] bits 1-0, fixed register reference to the View Resizer Vertical Scaling Rate bits which should be “REG[244Ch]” instead of “REG[244Eh]”
- REG[1808h] bits 2-0, changed note 6 to “For 8bpp operations the BitBLT must begin on an even boundary and the width must be an even number of pixels.”
- REG[3010h] ~ REG[3012h], removed references to 32 bpp mode from the bit descriptions
- section 11.1, updated the Power-on/off figure to include “Initialize Registers” in Normal Mode
- section 14.3.4, wording update for clarity
- section 20.4, updated paragraph 2 to clarify when the Host interface can return data without a new memory read access
- section 20.4, updated the list of conditions that will clear the memory read buffer
- section 20.4, added a paragraph about the memory buffer control bit
- section 20.4, for the note clarified the procedure to avoid coherency issues for both direct and indirect interfaces
- section 20.5.1, changed the maximum frequency for the Host Serial Clock to 16MHz
- section 22.1, added missing note regarding transfer count to the YUV Capture from Memory figure